107, *Formalization of Programming Concepts*, pp. 1–43. Springer, New York (1981).

3. R. M. Burstall and J. Darlington, A transformation system for developing recursive programs. *JACM* **24** (1), 44–67 (1977).
4. R. M. Burstall and J. A. Goguen, Putting theories together to make specifications. *Proc. 5th IJCAI, Boston*, pp. 1045–1058 (1977).
5. J. Darlington, *The Synthesis of Implementations for Abstract Data Types*. Research Report, DOC 80/4, Department of Computing, Imperial College, London (1980).
6. J. Darlington, Program transformation. In *Functional Programming and its Applications*, edited J. Darlington, P. Henderson and D. A. Turner. Cambridge University Press (1982).
7. R. W. Floyd, Algorithm 245 treesort. *CACM* **7** (12) (1964).
8. J. V. Guttag, Abstract data types and the development of data structures. *CACM* **20**, 397–404 (1977).
9. P. G. Harrison and H. Khoshnevisan, *On the Synthesis of Inverse Functions*. Research Report, Department of Computing, Imperial College, London (1988).
10. C. A. R. Hoare, Proof of correctness of data representations. *Acta Informatica* **4**, 271–281 (1972).
11. R. B. Kieburtz and J. Shultis, Transformations of FP program schemes. In *Proceedings of ACM Conference on Functional Languages and Computer Architecture, Portsmouth, New Hampshire* (1981).
12. B. H. Liskov and S. N. Ziller, Specification techniques for data abstractions. *IEEE Transactions on Software Engineering* **SE-1** (1), 7–19 (1975).
13. J. Vuillemin, *A Data Structure for Manipulating Priority Queues*. Internal Report, Département d'Informatique, Université de Paris Sud, France (1976).

Book Reviews

GERHARD BREWKA
Nonmonotonic Reasoning: Logical Foundation of Commonsense
Cambridge University Press. £19.50.
ISBN 0 521 38394 3.

This book discusses aspects of nonmonotonic reasoning at three levels: logic (proof- and/or model-theoretic accounts of nonmonotonic systems), implementation (automated theorem provers for nonmonotonic systems) and what the author terms pragmatic (rule-based nonmonotonic systems or nonmonotonic process systems).

It is at the first level where most of the influential approaches to nonmonotonic reasoning have been proposed. The book begins with a broad overview of many of these approaches. Some of the prominent features are (1) the clarity with which what many Artificial Intelligence (AI) workers consider to be difficult approaches are presented; and (2) the adequate accuracy of the presentation, which seems to have done justice to every approach cited in the book up to the point of controversy. Therefore, the book should prove extremely useful and valuable for the novice who needs an almost painless introduction to a very complicated area of research.

As for the researcher who is already in the field the book offers less, but it is still adequately valuable and interesting. Undoubtedly, there are some new ideas such as the author's proposal of preferred subtheories which are both stimulating and beneficial to a researcher.

The second level, to which only one chapter is dedicated, is as interesting as the first. The computational aspects of nonmonotonic logics are interestingly difficult to handle. More is expected at this level if one of the main objectives of the book is to show that theoretically sound nonmonotonic reasoning can be done efficiently and if the author believes that 'there will probably not be much progress in the development of formalisations, nor an increase in the trust in the existing ones, without programs, which can handle more realistic examples than those which have been studied so far'. It would have been extremely helpful if the book could have given or, at least, emphasised the need for an in-depth comparison of the different approaches cited at level one in terms of their computational properties.

For the pragmatic level, which is the most appealing to AI workers, the book presents an interesting attempt at formalising some type of 'truth maintenance systems' by providing model theoretical semantics.

The overall expository value of the different approaches and different levels of the book cannot be under-estimated. The book has been fairly successful in its main goal, which is to give a broad overview of the state of the art in different fields of research in the area of nonmonotonic reasoning. It does not give more than a vague orientation in the field. It regards the different research activities in the field as alternative ways of achieving sound theoretical foundations and efficient computation of nonmonotonic reasoning. This, in fact, is what one ideally wants to happen and believe, but the reality is something different. As earlier AI literature shows, workers at the pragmatic level had for a long period dismissed the idea of formalisation and sound theoretical foundation. What is surprising is that the book's message is that different activities, rather than different approaches to nonmonotonic reasoning, are ways of achieving a common goal.

N. OBEID
Colchester

DAVID LIGHTFOOT
Formal Specification using Z, Macmillan.
ISBN 0-333 544080. £13.99

The zealots of the 'formal methods' community will not like this book: it contains more natural language than dense mathematics, was not written using LaTeX, and is an attempt to teach a formal notation rather than show off the author's mathematical ability.

Lightfoot has concentrated on the system specification aspects of Z and has wisely ignored attempting to teach development, a topic which generates a large amount of concrete mathematics – even when the development is simple, such as the transformation from a sequence to a linked list.

The style of the book is pleasantly relaxed and the pace is not too fast. The author covers the important 80% of Z relevant to system specification in the traditional way: starting with logic and progressing, via sets, to functions and sequences. The only weakness of the book is that it eschews large examples. One of the major advantages of Z is its structuring facilities, particularly the way that the schema calculus enables system specifications to be presented in an incremental way. The author would have improved the book if he had included a final chapter describing a substantial example.

The potential audiences for this book are students on degree and HND courses, and staff in industry with a little mathematical knowledge who want a quick introduction to formal methods. Certainly, this is the first formal methods book that I have read which is suitable for HND students – albeit students in the final year of their course.

The formal methods community – and I count myself a member of it – seems to be in crisis. For the last ten years it has attracted quite a large amount of funding and formal methods are on the syllabus of every university computing department; yet progress has been pitifully slow, with the occasional embarrassing failure such as the Viper microprocessor acting as a punctuation mark. Certainly, the vibrations that I perceive from the Department of Trade and industry and the Science and Engineering Research Council are of increasing dissatisfaction with the subject over its lack of progress.

One of the reasons for this lack of progress is the small amount of good technology transfer materials that has been produced. The formal methods community has, over the last decade, contented itself with writing postgraduate-level textbooks and using them to teach undergraduates. Happily, this seems to be no longer true. The recent issue of *An Introduction to Formal Specification and Z* by Till, Potter and Sinclair, published by Prentice-Hall, has started to reverse the trend. This book continues it. The problem though, I suspect, is that it has been written five years too late.

D. INCE
Milton Keynes

Downloaded from https://academic.oup.com/comjnl/article/35/2/147/360249 by guest on 09 April 2024