

References

- BOHN, C., and JACOPINI, G. (1966). Flow diagrams, turing machines, and languages with only two formation rules, *CACM*, Vol. 9, No. 5, pp. 366-371.
- CLARK, B., and HORNING, J. (1971). The system language for project SUE. *SIGPLAN Notices*, Vol. 6, No. 9, pp. 79-88.
- DIJKSTRA, E. W. (1968). Goto statement considered harmful, Letter to the Editor, *CACM*, Vol. 11, No. 3, pp. 147-148.
- FELDMAN, J. (1966). A formal semantics for computer oriented languages and its application in a compiler-compiler, *CACM*, Vol. 9, No. 1, pp. 3-9.
- GRIES, D. (1969). Compiler implementation language, *SLAC Report No. 102*, Stanford Linear Accelerator Center, Stanford, California.
- HOPKINS, M. (1972). A case for the goto, *Proceedings ACM National Conference*, August 1972.
- LEAVENWORTH, B. (1972). The goto controversy, *Proceedings ACM National Conference*, August 1972.
- MCAFEE, J., and PRESSER, L. (1972). An algorithm for the design of simple precedence grammars, *JACM*, Vol. 19, No. 3, pp. 385-395.
- PRESSER, L., and WHITE, J. R. (1972). Linkers and Loaders, *Computing Surveys*, Vol. 4, No. 3, pp. 149-167.
- PRESSER, L. (1972). The translation of programming languages, in *Computer Science*, Cardenas, Presser and Marin (eds.), John Wiley and Sons, New York.
- WHITE, J. R., and PRESSER, L. (1973). A tool for enforcing system structure, *Proceedings ACM National Conference*, Atlanta, Georgia, August 1973.
- WIRTH, N. (1971). The programming language PASCAL. *Acta Informatica*, Vol. 1, No. 1, pp. 35-73.
- WULF, W. (1972). A case against the goto, *Proceedings ACM National Conference*, August 1972.

Book reviews

PL/I in easy stages: A programmed learning textbook, by J. S. Roper, 1973. (*Elek Science*, £2.00.)

The aim of this book is to give a student a good working knowledge of the fundamental aspects of PL/I. It represents a programming course for a subset of PL/I, which has been taught to students from different disciplines at Durham University. As the title implies, the material is organised in the form of a programmed learning text. The course is divided into a list of topics. Each topic is presented in a section with three parts. Basic material about a topic is given first. If this is not 'understood fully' by the student, then he must read the expanded material on the same topic and answer the test questions that follow.

It is not at all clear if this book is intended to be a self-study text. If this is the case, then a significant omission is some statement on the background which the student or reader should have. The topics included and the order in which they are presented, do suggest a FORTRAN-based approach. ALGOL-like concepts, such as, blocks and scope of variables, are discussed towards the end of the book. In addition, some comments made would be meaningful only to those with a FORTRAN background, for example, the note—'Until a later chapter covering declarations has been read, the beginner is advised to avoid choosing identifiers beginning with any of the letters I to N inclusive'.

The presentation of topics does not appear to be well-structured. For example, arithmetic expressions are discussed (in Chapter 1) before any firm foundation in or understanding of the basic data representations and attributes is given (in Chapter 3). The assignment statement is explained in the section on arithmetic expressions, whereas the conditional expression is explained in the section on branching statements (in Chapter 2).

The treatment of many topics is brief and superficial and much emphasis is placed on the variations in syntax rather than the seman-

tic implications. Further, there is an absence of consideration for the relative importance of the many concepts covered. The brevity of treatment is reflected by the fact that the basic material for most of the topics is presented in four pages or less, in this 5½ by 8½ inch book. These topics include on-statements, arrays, blocks and storage allocation, and structures.

Finally, the programs used as examples are for simple problems and can be easily understood. However, they reveal a fondness for the GO TO statement. This is regrettable since opportunities to demonstrate alternative constructs are missed.

A. L. LIM (London)

Applied Finite Mathematics, by H. Anton and B. Kolman, 1974. 475 pages. (*Academic Press Inc*, \$10.95.)

This is an elementary text book which starts with an introduction to sets, co-ordinate geometry and matrices; it then proceeds to introduce linear programming using first a geometric approach and then an algebraic one. A chapter introducing basic ideas in probability is followed by another on statistics including standard distributions and tests of hypothesis.

Although the book starts from an elementary level, it covers topics such as duality in linear programming, Bayes' formula in probability, Chebyshev's theorem in statistics. There is a chapter on applications in medical diagnosis, genetics, game theory and Markov chains. The final chapter introduces computers and programming in FORTRAN.

The book is well produced and liberally illustrated with diagrams in colour, the results are clearly displayed, there is a large number of exercises with answers. It is strongly recommended for first year College and University students in the social and biological sciences.

I. M. KHABAZA (London)