

3. the serial connection of $M_{\phi'}$ and $M_{\phi/\psi}$ is a machine $M_{\psi'}$ for cover ψ' with function λ^* mapping the states of $M_{\psi'}$ onto the set ψ' .

(Thus $M_{\psi'}$ replaces M_{ψ} and, ψ' replaces ψ in the definition of λ^* .)

ERIC G. WAGNER

IBM Thomas J. Watson Research Center
Yorktown Heights, New York 10598

1. K. B. KROHN & J. L. RHODES, *Algebraic Theory of Machines*, Proc. Sympos. Math. Theor. Automata, New York, April 25-26, 1962, *Microwave Research Institute Symposium Series*, Vol. XII, Polytechnic Press, Brooklyn, New York, 1963.

2. H. P. ZEIGER, *Loop-free Synthesis of Finite State Machines*, Ph.D. Thesis, Massachusetts Institute of Technology, Cambridge, 1964.

3. ANN PENTON, *Algebraic Study of Sequential Machine Decomposition*, Master's Thesis, Wesleyan University, Middletown, Conn., 1967.

25 [12].—FRANK BATES & MARY L. DOUGLAS, *Programming Language/One*, Prentice-Hall, Inc., Englewood Cliffs., N. J., 1967, viii + 375 pp., 25 cm. Price \$5.95.

This book is a welcome addition to the literature of PL/I. It is written in a clear and concise style covering a wide field. In spite of the uncertainties about how the PL/I language will finally be implemented (witness the frequently changing specifications used by the manufacturer); this book manages to convey an idea of the power of the PL/I language and to develop in the reader a facility for writing clear, efficient PL/I code.

The examples are easily understood and to the point. Several of the problems provide good practice in the fine art of debugging. The answers seem to be correct and complete.

A very fortunate feature of this book is that technical points (e.g., the inaccuracy caused by representing a decimal fraction in a base other than ten) are reserved until the end of the appropriate chapter, where they appear in sets of notes. This is commendable, since it provides the reader with useful technical information without disturbing the flow of the more basic material.

The book contains several useful appendices, including PL/I character sets, keywords and abbreviations, built-in functions, conditions and format specifications. These tables, combined with a thorough index, make this book valuable as a reference work for the experienced programmer as well as useful as an introductory text to the subject.

TOM KEENAN
HENRY MULLISH

New York University
Courant Institute of Mathematical Sciences
New York, New York 10012

26 [12].—CHARLES R. BAUER, ANTHONY P. PELUSO & WILLIAM S. WORLEY, JR., *IITran/360: Self-Instructional Manual and Text*, Addison-Wesley Publishing Co., Reading, Mass., 1967, xi + 212 pp., 28 cm. Price \$4.95.

This book is an introduction to a new breed of computer languages called IITRAN, an acronym for Illinois Institute of Technology Translator, a language

greatly resembling PL/I and one which processed all of the worked examples on an IBM 360 Model 40.

Unlike so many other books in the field, this text does not assume a high level of familiarity with computer languages, and the exercises seem to be well geared to introduce the novice to the basic notions without his having to seek the assistance of outside help. The ideas are presented clearly and a variety of techniques is employed, thus making the book both informative and pleasant to read—a most unusual combination.

The contents of the book could be thoroughly digested in a period of only a few hours, and the reader should be able to write a computer program after a short while. But, as the authors state in the preface, a mastery of the material covered will not transform one into a “programmer.” Considerable practice and experience will, indeed, be necessary before proficiency can be attained.

HENRY MULLISH

New York University
Courant Institute of Mathematical Sciences
New York, New York 10012

27 [12].—CHARLES PHILIP LECHT, *The Programmers ALGOL: A Complete Reference*, McGraw-Hill Book Co., New York, 1967, xxiii + 251 pp., 28 cm. Price \$8.95.

This book might be more appropriately called an elementary primer. Like a primer, it has large type, excessive white space, “controlled” introduction of material and incessant repetition. Unlike a “complete reference” it lacks an index and a reasonable glossary (it has an unreasonable glossary of 19 entries).

According to the preface, the book was derived not from the official report of the ALGOL authors, but from the manual for the GE 625/635 ALGOL compilers. ALGOL coding is printed in a hardware representation rather than in the reference language. About 30 percent of the book is devoted to one manufacturer’s approach to input/output. Of the seven primitive I/O procedures recommended by IFIP/WG 2.1, only three are mentioned.

Although the author claims the advantages of basing his work on an actual compiler, he frequently fails to clarify what position “his” compiler takes on well-known ambiguities in the ALGOL report. He attempts to make ALGOL easier to swallow—cutting it up into bite-size pieces by unwinding some of the recursion in the definitions—a format which requires much repetition. As a result, some rules have been stated in a short form which is harder to take than the original official formulation. For example, it takes some ten pages to get to the description of statements of the form: *if b_1 then s_1 else if b_2 then s_2 else . . . if b_{n-1} then s_{n-1} else s_n .*

For the most part, it is not easy to find in this book any discussions of the subtler or more uncertain points of ALGOL such as those raised by Knuth and Merner in “ALGOL 60 Confidential” or those explicitly left unresolved in the revised ALGOL report (which of course had to be resolved somehow in the compiler on which the book is based).

Some important omissions are the following:

(a) No mention is made of the initial values of *own* variables.