reading for people without a strong mathematical orientation. This main development is supplemented by a number of self-contained and more expository papers which approach the semigroups from a variety of more machine-oriented points of view. This includes Zeiger's independent proof of the decomposition theorem.

The final two papers show two approaches to context-free and other languages: a grammatical approach and a power series approach.

The somewhat specialized subject matter of the book and its diversity of notation, level, and style make it an unlikely choice as a textbook, but it could prove valuable as a reference book and most people with interest in automata theory are likely to find some of the material of interest. Its strongest virtue is in its diversity of approaches and viewpoint.

R. E. Stearns

General Electric Research and Development Center
Schenectady, New York 12301

54[12].—D. W. Barron, *Recursive Techniques in Programming*, American Elsevier Publishing Co., Inc., New York, 1968, 64 pp., 22 cm. Price $5.25.

This monograph deals with the use of recursive techniques in programming. It considers very briefly and sketchily the ideas of recursion, the mechanisms for implementation and the formal relationship between recursion and iteration. It also contains some examples and applications. It is the sort of material that should be covered in a lecture or so in an introductory course in computer science.

M. Goldstein

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

55[12].—Peter Wegner, *Programming Languages, Information Structures and Machine Organization*, McGraw-Hill Book Co., New York, 1968, xx + 401 pp., 23 cm. Price $10.95.

This book is very much a mixed bag. On the one hand, it is a useful collection of information on many of the most modish topics in computer science today; on the other hand, it suffers from a general lack of organization, occasional narrowness of viewpoint, and sometimes obscure explanations. The author states in his introduction that he plans ". . . to classify programming techniques and to develop a framework for the characterization of programming languages, programs, and computations. In the present text such a framework is developed, starting from the notions that a program with its data constitutes an *information structure*, and that a computation results in a sequence of information structures generated from an *initial representation* by the execution of a sequence of *instructions*." In fact, it appears that these notions are too abstract to be applied in any meaningful way to the subject matter of this book.

The first chapter of the book is on machine language and machine organization. The author treats the instruction set of the IBM 7094 as a paradigm for computer