

University of Calgary
Calgary 44, Alberta, Canada

1. RICHARD K. GUY & J. L. SELFRIDGE, "Interim report on aliquot series," *Proc. Winnipeg Conf. on Numerical Math.*, October 1971.
2. G. AARON PAXSON, "Aliquot sequences" (preliminary report), *Amer. Math. Monthly*, v. 63, 1956, p. 614.

39[12].—B. A. GALLER & A. J. PERLIS, *A View of Programming Languages*, Addison-Wesley Publishing Co., Reading, Mass., 1970, vi + 282 pp., 24 cm. Price \$12.95.

Alan Perlis and Bernard Galler have been major contributors towards the design of programming languages. In this book, they present their views on the structure of programming, and describe a programming language, Algol D, that reflects these views. The book would be better entitled *A View of a Programming Language*, since they make little attempt to describe or deal with realistic programming languages other than Algol D. Thus, problems such as input-output and parallel processing are barely mentioned, and the reader may have a hard time seeing the connection between the ideas of this book and his favorite programming language. Nevertheless, the book must be regarded as a significant contribution to its field.

The expository quality is excellent. There are many exercises, and these are not only related to the text but are actually referenced within it. Thus, the reader who does the exercises will have extended the material covered by the book. The presentation is quite well-organized and ought to be suitable for a variety of readers, though some acquaintance with programming is surely necessary. Although the book is intended for classroom use as well as self-study, an instructor with well-formulated views on programming languages would probably not feel comfortable teaching from it because of its strong and rather personal biases.

There are four chapters. The first chapter is an elegant development of a higher-level (but not really practical) programming language starting with Markov algorithms. With the simplest type of algorithm as a base, the concepts of concatenation of algorithms, subroutines, operation on part of the data space, labelling, and storage addressing are introduced. The result of these successive extensions is the Addressed Labelled Markov Algorithm (ALMA). Emphasis is placed on the use of conventions in order to represent different data structures with the same set of characters. This chapter also introduces the concept of interpretation.

The second chapter is concerned with language. It begins with a discussion of flowcharting, and then moves on to present a version of Algol without arrays. Arrays are then presented in the context of a more general development of data structures, including strings as well. Unfortunately, the section on flowcharts contains a misleading explanation of an algorithm for finding a zero of a continuous function in an interval $[p, q]$. Although the description of the algorithm specifies that the interval must contain an odd number of zeros, the algorithm itself does not assume this. Under the stated assumption, the first half of the algorithm is superfluous.

The third chapter is concerned with data structures. In this chapter, three basic structure-forming operations are introduced, one for strings, one for arrays, and one for nonhomogeneous n -tuples. Since names (i.e., locations) of data are also allowed

as data, it becomes possible to build list structures of various kinds. A number of examples of different data structures are given, including multidimensional arrays, complex numbers, ring structures such as those used in graphics, and threaded lists. The notion that complex structures may be built up from simple ones through a hierarchy of definitions is emphasized.

In the final chapter, concerned with extensible languages, the authors discuss the incorporation of definitions into Algol in order to produce Algol D. The definitional method deals not only with the specification of the structures themselves but also with the specification of operators on the structures. Since the same operator may have different meanings when applied to different structures, the process of determining the meaning of an operator is context-dependent. Given a context, it is then possible to invoke an appropriate substitution of text in order to replace an occurrence of a defined operator by its definition. The process of context determination and replacement is discussed in some detail. The book concludes with a discussion of macro schemes suggested by other people.

In summary, this is a worthwhile and meaty book, though one that presents a highly personal and limited view of its field.

PAUL W. ABRAHAMS

Courant Institute of Mathematical Sciences
New York University
251 Mercer Street
New York, New York 10012

40[12].—RONALD BLUM, Editor, *Computers in Undergraduate Science Education*, Commission on College Physics, College Park, Maryland, 1971, x + 499 pp., 23 cm. Available from American Institute of Physics, 335 East 45th Street, New York, New York 10017.

These conference proceedings contain quite a number of papers dealing with the use of computers in college physics instruction, as well as some dealing with broader aspects of the use of computers in universities. None of the material here is of lasting significance, though the book has a certain pragmatic value for those actively engaged in educational projects using the computer as a tool. The numerous reports on active projects and their results do provide a standard of comparison for workers in the field. For other readers, the book may be worth skimming as a matter of interest but does not merit detailed study.

Fortunately, the authors of most of the papers seem to have tried to present their results honestly without exaggerating their successes. I was rather charmed by an article by Edwin F. Taylor entitled "History of a failure in computer interactive instruction," which begins with the sentence "This paper deals with an elegant and technically successful computer interactive display that has not influenced many students."

The first groups of papers discuss applications where the computer is used by students as an experimental tool. Some of these applications require the student to know how to program, while others do not. The simulation of the behavior of physical