References

Barraclough, E. D. (1965). The application of a digital computer to the construction of timetables, *The Computer Journal*, Vol. 8, p. 136.

Clementson, A. (1968). Computer Timetabling Data Manual. Published by the Local Government Operational Research Unit.

Cox, N. S. M. (1969). Final Report on the Research Project into the Construction of Timetables by Automatic Computer. To be published by the University of Newcastle upon Tyne Computing Laboratory, 1969.

Gilmore, P. C., and Gomory, R. E. (1961). A Linear Programming Approach to the Cutting-Stock Problem, *Operations Research*, Vol. 9, No. 6, p. 849.

Gomory, R. E. (1958). An Algorithm for Integer Solutions to Linear Programs, *Recent Advances in Mathematical Programming*, ed. R. L. Graves and P. Wolfe, McGraw-Hill, New York, 1963.

Johnston, H. C., and Wolfenden, K. (1968). Computer Aided Construction of School Timetables, paper presented at IFIP Conference.

Lawrie, N. L. (1968). School Timetabling by Computer, *Aspects of Educational Technology*, Vol. II, Methuen, London, 1968.

Lewis, C. F. (1961). *The School Timetable*, Cambridge University Press.

Lions, J. (1967). Construction of Timetables for Ontario Schools Using a Computer, O.E.C.D. paper, Paris.

Local Government Operational Research Unit (1967). The Use of Computers for School Timetabling. Report No. C19. Reading.

Book Review

The Art of Computer Programming, by Donald E. Knuth Volume 1—Fundamental algorithms. 1968. 634 pp. 182s.; Volume 2—Seminumerical algorithms. 1969. 624 pp. (*Addison-Wesley*, 173s.)

These two heavy volumes are the first to be published of a series of seven forming a work with the general title 'The Art of Computer Programming'. When completed, this work will be an immense achievement. It is impossible to do justice even to the first two volumes in a short review, and it would be presumptuous to attempt a full evaluation without giving them a long and detailed study. I can only attempt to define the scope of the work and its flavour, and to make one or two scattered comments.

The books are addressed to people who approach things from the mathematical point of view. Such people enjoy proofs and like to be set puzzles. They will find plenty here. Some 25% of Volume 1 is taken up either in setting problems or in indicating their solutions. The author has taken the trouble to classify the exercises according to their difficulty and degree of mathematical orientation.

There is no doubt that, as well as being a mathematician, Dr. Knuth is a practical man when it comes to dealing with a computer. He discusses programming down to the level of assembly language, and uses for this purpose an imaginary computer known as MIX, whose assembly code is fairly typical of real assembly codes. He does not attempt to cater for the complete beginner, but he does give a comprehensive treatment of the finer points. I was delighted, for example, to find the first comprehensible account that I have seen of co-routines (although we shall have to wait for Volume 4 for a treatment of recursive co-routines). Nevertheless, the fact remains that the overwhelming, almost overbearing, atmosphere of the book is one of mathematics. It would be unfortunate if some ordinary mortal, attracted by the title and charmed by the style, were, nevertheless, led to conclude that he needed a high standard of mathematical knowledge in order to understand programming.

The first 119 pages of Volume 1 are about general mathematics and could just as well have appeared in a book on, for example, quantum mechanics. After that, the author goes on to describe his imaginary computer MIX, and in terms of it to discuss fundamental programming techniques such as subroutines, interpretive routines, trace routines, etc. He then discusses information structures, including stacks, queues, linked lists, trees, and so forth. However, it is not long before he passes from programming to mathematical problems suggested by programming, such as the enumeration of trees.

Volume 2 deals with random numbers and with algorithms for performing arithmetic operations. It is stated in the preface to contain a noticeably higher percentage of mathematical material than other volumes in the series. Methods of generating a series of pseudo-random numbers in a computer are exhaustively discussed. It is pointed out that some methods are better than others, and statistical techniques for evaluating them are developed.

In the next chapter, the author goes systematically through floating-point arithmetic, multi-precision arithmetic, radix conversion, rational arithmetic, and polynomial arithmetic. The material partly concerns the designer of the algorithms that are wired into a computer, and partly the designer of subroutines or procedures that are incorporated within programming languages. These subjects are of basic importance, and much time and money has been wasted because people have not understood them properly.

Here and there in the volumes there are historical notes. These, like everything else, are flavoured with the author's personality; they are adequate for their purpose, but not necessarily definitive from an historian's point of view.

There can be few people in university computer departments who will not get some value and pleasure from browsing in these volumes. Those who are able to learn by studying details, and who like to follow up interesting side issues, will read them systematically. Others will perhaps recall the story of the man who apologized for writing a long book, saying that, unfortunately, he could not spare the time to write a short one. However, Dr. Knuth, generous with his time as with his scholarship, has foreseen this criticism and plans to publish later a shortened version of the series. I hope that in preparing this he will have the non-mathematician in mind.

M. V. Wilkes (Cambridge)