

where $P(z)$ is a polynomial of degree n , $k \simeq n/2$, $Q(z)$ is a polynomial of degree $n - k$, and $R(1/z)$ is a polynomial of degree k . He claims that 'instead of evaluating $P(z_0)$ we evaluate $Q(z_0)$ and $R(z_0)$ where in each case the round off error will be considerably less than for $P(z_0)$ '. The method of evaluation is not specified, but if Horner's rule is used it is easy to see that the computed value of $Q(z_0) + R(z_0)$ will be the exact value of $\tilde{Q}(z_0) + \tilde{R}(z_0)$, where $\tilde{Q}(z)$ and $\tilde{R}(1/z)$ are the same polynomials as $Q(z)$ and $R(1/z)$ except that their coefficients \tilde{q}_i and \tilde{r}_i have suffered slight perturbations (in fact

$$|\tilde{q}_i - q_i| = O(n\epsilon |q_i|)$$

if ϵ is the machine precision). But the same is true if we simply evaluate $P(z_0)$ and then divide by z_0^k (Wilkinson, 1963)! Westreich does not give any convincing theoretical or empirical evidence that his method is more accurate than the obvious one. Whether Westreich's suggestion is used or not, the only difficult decision is whether $P(z_0)$ (or $F(z_0)$) is 'sufficiently near zero' or not, but Westreich does not give any criterion for making this decision.

References

BRENT, R. P. (1973). Reducing the Retrieval Time of Scatter Storage Techniques, *CACM*, Vol. 16, pp. 105-109.

GONNET, G. H. and MUNRO, J. I. (1979). Efficient Ordering of Hash Tables, *SIAM J. Comput.*, Vol. 8, pp. 463-478.
MADDISON, J. A. T. (1980). Fast lookup in hash tables with direct rehashing, *The Computer Journal*, Vol. 23, pp. 188-189.
MALLOCH, E. S. (1977). Scatter storage techniques: A unifying viewpoint and a method for reducing retrieval times, *The Computer Journal*, Vol. 20, pp. 137-140.
WESTREICH, D. (1980). Improving polynomial evaluation at an approximate root, *The Computer Journal*, Vol. 23, p. 187.
WILKINSON, J. H. (1963). *Rounding Errors in Algebraic Processes*, HMSO, London, Ch. 2.

Yours faithfully,

R. P. BRENT

Department of Computer Science
Australian National University
Box 4, Canberra, A.C.T. 2600
Australia
14 August 1980

Book reviews

Theory and Design of Digital Computer Systems 2nd ed. by Douglas Lewin, 1980. (Nelson, £8.95)

This is a second edition of the same author's *Theory and Design of Digital Computers* (Nelson, 1972) and has been produced largely by the insertion of new paragraphs and sections, with very few deletions: in fact the only significant deletions are the sections on magnetic thin film stores and delay-line storage systems; in their place we find a greatly expanded section on semiconductor storage including charged coupled storage systems and magnetic bubble memories. By far the greatest distribution of new material is that relating to the new LSI devices, including microprocessors, but the largest single insertion is in the chapter on highly parallel processing systems.

The method of editing employed no doubt saved both author and publisher considerable effort, and at a first reading the seams hardly show. However, it does explain the continued existence of such errors as 'principle' used as an adjective and 'Oersteds' instead of 'Oersteds'. The latter is just one example of non-SI units which should have been removed; another is 'mils' which the UK reader may easily misunderstand as 'millimetres' though it means thousandths of an inch, smaller by a factor of almost forty! There also remain statements which were hardly still true in 1972, such as 'Numbers are generally represented in the computer using fixed-point binary fractions', 'For representing instructions . . . the octal system . . . is used', 'multiple-level indirect addressing . . . is incorporated in many commercial computers' and 'Data are normally recorded on the magnetic tape one 6-bit character . . . at a time . . . using seven recording channels', which have nevertheless escaped revision. If I recommend this book to my students, what will they make of integer number representation, the growing use of hexadecimal notation, my assertion that the last commercial computer to use multilevel indirect addressing was the ill-starred BCL Molecular 18, and my reference to seven channel magnetic tape as obsolescent? True, hexadecimal notation is also described, but the justification given for it is the eight-bit byte with the PDP-11 as the only specific example of such a computer—yet all the PDP-11 literature uses *octal* notation, not because of its eight-bit byte but because of its eight general registers and eight addressing modes! Surely the IBM 360 (or 370) with its 16 general registers would have been a more convincing example?

Some dated terminology persists, like 'order' for 'instruction' and 'modifier register' for 'index register'. The use of papertape, more-over five-hole paper tape, for introducing data into the computer will seem strange to present day students. 'The standard teletype' is now surely Model 43 and not the Model 33 whose keyboard is illustrated, and the present day student is in any case more likely to use a 96-character VDU. Perhaps Pascal or ALGOL 68 should have replaced the ALGOL 60 (here simply called 'ALGOL') example; I ignored the non-ALGOL 60-like syntax of the I/O statements because some older implementations did have such I/O; but on seeing non-

standard FORTRAN I/O I looked up the reference and found that it was to FORTRAN II and dated 1962. What about 1966 ANSI FORTRAN, let alone FORTRAN 77? One could go on nit-picking, yet I shall continue to recommend 'Lewin' to my students and to all who want a good grounding in computer hardware at a level between that of a monograph and a fullscale computer engineering course. In fact I hope that this second edition will sell so well that a thorough revision will be made for the third edition in the mid-1980s.

D. J. CAIRNS (London)

Statistical Computing, by W. J. Kennedy and J. E. Gentle, 1980; 591 pages. (Marcel Dekker, SF58)

This book is a compendium of what every statistician should know about the numerical side of computing. It does not cover any non-numerical techniques such as data structuring, file handling, sorting, and graphical output that figure largely in the work of many statisticians, but within this limitation it is thorough and comprehensive.

The book starts with an admirably clear and careful account of computer arithmetic, and continues with chapters on calculation of probabilities and percentage points of standard distributions, generation of pseudo-random numbers, matrix techniques, multiple regression (including stepwise but not ridge, for some reason), analysis of variance, unconstrained optimisation, multivariate analyses and criteria other than least squares. Within each chapter, the numerical analysis 'theory' is presented tersely as a peg on which to hang brief accounts of how it has been implemented as algorithms. The references are a major part of the whole; the authors must have achieved near completeness and the reader is never short of things to look up if more details of any technique are needed. There are, for example, 323 references in the chapter on pseudo-random numbers alone. I wonder, though, if such completeness is worthwhile, as many papers seem to be mentioned almost just for the sake of it. A more selective list of key papers would have been more helpful, as would more specific mention in the text of where algorithms to do particular tasks can be found. Having read up what a Cholesky decomposition is, for example, I then had to plough through the references at the end of the chapter to save myself the job of writing my own algorithm. There are, however, quite a few algorithms given in machine independent flowchart form in the text that are clear and easy to code.

I shall be glad to have this book for occasional reference and shall certainly use parts of it in teaching statistics courses at all levels. I shall not base a course on it as the terse style and unattractive layout (photographed typescript) seem likely to make it unappealing to students. The material is very much the kind of thing we should be teaching more of, especially at M.Sc. level, in the UK, though.

P. R. FREEMAN (Leicester)