KOBAYASHI, H. (1974). Application of the Diffusion Approximation to Queuing Networks, I: Equilibrium Queue Distributions, *JACM*, Vol. 21, No. 2, pp. 316-328.
MITRANI, I. and HINE, J. H. (1975). Complete Parameterised Families of Job Scheduling Strategies, Technical Report 81, Computing Laboratory, University of Newcastle upon Tyne, October 1975.
SALTZER, J. H. (1974). A Simple Linear Model of Demand Paging Performance, *CACM*, Vol. 17, pp. 181-185.
SALMAN, O. (1976). Mathematical Modelling of Computer Systems, Ph.D. Thesis, University of Nottingham.

Book reviews

Managing Software Projects, by J. K. Buckle, 1977; 108 pages. (*Macdonald and Janes*, £3·95)

This is a slim volume in the series of 'Computer Monographs' published by Macdonald and Jane's, written by British authors, mainly I suspect with experience in one way or another of ICL machines and methods. Mr Buckle, who uses his experience in ICL for the basis of his 108 pages, does provide some good ideas on writing software to a planned budget for anyone who is totally new to the subject. But for a prospective reader who has experienced the frustrations of actually trying to control large scale program development, he provides little that is new or concrete and tends to make do with vague generalisations in the really tricky areas.

The author defines software as being not only computer manufacturers' compilers and operating systems, but also a user's implementation of a complex data base; or even a suite of programs based on a single set of files where intercommunication is important. But one has the feeling that his heart is really in writing the manufacturer's software and the environment he describes and the methods he proposes seem less suited to a normal DP installation—see the discussion on 'Releases', 'Phasing' and 'Implementation Tools'.

Mr Buckle does not seem to cover the function of the systems analyst and omits his investigations and work on economic justification, in order to concentrate on the job of the project leader of the programming team. This project leader tends to be depicted not only as all powerful in controlling the resources, but to be given the task of producing software without any real need to justify the project or its total cost. In the real world of commercial DP, estimates of programming cost and time have to be made at the initial proposal stage, long before the task is well enough defined to suit the techniques of Mr Buckle's project leader. If the project is given the go-ahead, then it is these initial estimates which tend to form the project leader's budget—whatever his subsequent investigations may indicate. Also the project leader has to compete for programmer resources with all the other projects and suffers from staff leaving and user departments changing their mind. The author says 'find a single person who can represent the user departments, and carry out all negotiations with him', but most 'customers' in firms would not be willing to entrust one person with their requirements, and desperately retain the freedom of individual bargaining. Even if one user by sheer willpower becomes the acknowledged negotiator, the odds are he will be promoted or transferred before the project is completed.

The author describes the typical documentation required to record and control the project, such as the 'Functional requirements specification', 'File descriptions', 'Module maps', 'Testing plans', 'Project log', etc. All these are of course worthwhile and one idea, the 'Slip Chart'—to show the different estimated times of completion at various stages of the project, measure slippage and record action—is a useful addition. But the crucial problem still arises as to what to do when there is slippage? What actions are possible if the estimates of programming time and complexity were wrong? What can be done if there are no spare programmers or less high priority projects, or no extra machine time available? How can you accurately measure progress? Mr Buckle is not very helpful, especially as he seems to rely on everyone acting logically, unemotionally and with great ability.

The difficult process of estimation of programming time and resources also receives little assistance in the book, reliance being placed on basing estimates on the number of lines of code per average programmer per unit of time. This seems to leave large scope for errors, especially in deciding how many lines of code there will be in the first place. Estimating techniques for software still seem to be primitive in most installations, including from this evidence, computer manufacturers.

Mr Buckle likens the development of software to the initial design and prototype construction of a car or a computer. This may be acceptable to 'customers' in computer manufacturers who will realise the problems of prototypes, but it will not be accepted by the average department manager of a business concern, who tends to be sceptical about using a computer system in the first place and needs reassurance that the program will really work rather than be asked to make allowances for the hazards of developing a new product. The computer profession must be able to do better than this or the potential benefits will never be accepted in principle nor obtained in practice.

To sum up, this is an important topic for the computer industry and the book is easy to read and worth perusing. But it cannot be recommended as the solution to the problem, least of all in commercial DP.

R. M. PAINE (London)

The State of the Art in Numerical Analysis, edited by D. A. H. Jacobs, 1977; 978 pages. (*Academic Press*, £20·00)

This book is based upon the proceedings of a conference of the same title organised by the Institute of Mathematics and its Applications in April 1976. The first residential conference of the IMA was held in 1965, also with the same title and the proceedings of that conference were also published in book form (*Numerical Analysis: an Introduction*, edited by J. Walsh, 1966; 212 pages. *Academic Press*). Dr Scriven notes in his foreword to the present volume that 'since the conference the use of computational methods to solve the mathematical problems arising from a great variety of application areas has exploded in keeping with the availability of increased computing power.' Thus, whereas the aim of the earlier volume was to provide 'a general survey of the principal topics of interest without attempting to cover them fully', the present 'volume surveys the theoretical and practical developments across a wide area of numerical analysis over the past decade', and fully justifies its title. The material in the book is presented in seven distinct parts which, with the exception of Part III which is the longest and contains five chapters, contain three chapters. The topics covered in these seven parts are I Linear algebra, II Error analysis, III Optimisation and nonlinear systems, IV Ordinary differential equations and quadrature, V Approximation theory, VI Parabolic and hyperbolic problems, VII Elliptic problems and integral equations; each of the 24 chapters is written by a specialist in the field. A feature of each chapter is the substantial number of references given. The book is handsomely bound and printed on good quality paper, although in the interests of economy the printed page is in the form of typescript photocopy. The price is probably sufficiently high to deter private subscription, but anyone who is interested in recent theoretical and practical developments in numerical analysis should ensure that he has access to a copy.

N. RILEY (Norwich)