need for skilled professional programmers to develop operating systems and to push the available equipment to the extreme limit of its capabilities, but their job is no longer the same as that of the programmer. If a programmer (or a man-with-a-problem, as he should perhaps be called) does feel the need to descend from his high-level language to an assembly language he will find in that language many features which are of very little use to him. They are only there for the benefit of compiler-writers and other professional programmers, and rightly so, since they should be the most frequent users of such levels of language. No argument is really as simple as this, and there may well be situations in which a problem-programmer needs detailed knowledge of his computer to avoid making inefficient use of it. This will still matter until computers are given away free in exchange for the wrappers from six input-output devices, and that will not be for a little while yet. The chances are that the best systems-programmers will be discontented problem-programmers, since they will be in the very best position to appreciate the users' real needs.

As a direct result of the use of operating systems it is possible economically to turn the man-with-a-problem into a problem-programmer able to communicate easily with an extremely powerful problem-solving tool. It may appear at the moment that he is being kept further away from his new tool by the operating system than his predecessor was. This may be true, but he is nevertheless as a user already getting better service from it, and all the hardware and software are now available to make it appear that each user has exclusive use of the system. It was a most fitting conclusion to the Edinburgh Conference that Dr. Wilkes should have shown this so clearly in his demonstration of the use of the MIT Compatible System. It is to be hoped that it will not take too long for the economic situation to allow us to put man's time at a higher premium than that of a machine.

References

BARRON, D. W., and HARTLEY, D. F. (1964). "The Influence of Automatic Programming on Machine Design", (*UKAC report*, p. 21).

CARR, J. W. III (1964). "The Future of Programming and Programmers," *The Computer Bulletin*, Vol. 8, No. 1, p. 9.

KILBURN, T., and GRIMSDALE, R. L. (1960). "A digital computer store with very short read time," *Proc. I.E.E.*, Vol. 107B, p.567.

PLAYFAIR, Sir EDWARD (1964). "Computers and psychology," *The Computer Journal*, Vol. 7, p. 1.

UKAC report on 1964 Edinburgh Conference: "The Impact of Users' Needs on the Design of Data Processing Systems," Organized by the B.C.S., Brit.I.R.E. and I.E.E.

WEGNER, P.—Editor (1964). *Introduction to System Programming*. Academic Press, London.

WILKES, M. V., RENWICK, W., and WHEELER, D. J. (1957). "The design of the control unit of an electronic digital computer," *Proc. I.E.E.*, Vol. 105, p. 121.

Book Review

Management Standards for Data Processing, by DICK H. BRANDON, 1963; 404 pages. (London: *D. Van Nostrand Co. Ltd.*, 93s.)

In the past few years, the literature of Data Processing has grown almost to a flood. Most of it has poured over the specialist—analyst, programmer, engineer—dealing with technical matters pertinent to his field. Much of the rest has been directed at managements, explaining the intricacies of the subject, and quieting their fears of its impact.

There has, however, been little of direct service to the harassed D.P. executive whose daily lot it is to turn the eccentricities of his wayward (if brilliant) brood into the targets achieved and work accomplished that alone can justify the existence of a computer installation. Mr. Brandon's book will do much to ease his sufferings. Clearly the content has been distilled from experience, and will prove to be an invaluable guide past the pits and traps which lie in wait for even the most experienced of those who seek to set up an efficient, productive, controlled data processing organization.

The theme of the book is Control, and its development is concerned with the establishing of standards to which every aspect of D.P. work can be subjected. To this end, all the software components (and a little of the hardware) are meticulously dissected, classified and coded, and their relationships with one another carefully defined. From the skeleton thus exposed Mr. Brandon builds up a series of suggested techniques by which the D.P. group can be started on the right path, and kept from wandering too far from it.

Systems Analysis, Programming, Operations Performance Evaluation are the major topics with which the book deals, and each is stuffed with formulae and numerical information. It would be unwise to apply the former to any specific case (other than the originating one) as they stand, but they do serve as a most useful starting point.

The essence of control in most organizations lies in adequate (but minimal) documentation, and it is heartening to read Mr. Brandon's constant iterations on this matter.

The book is specially recommended for systems analysts and programmers. They would then look with less disfavour upon the efforts of their superiors to guide their creative talents into narrower channels.

C. E. HARDING