worth noticing that FORTRAN I had a FREQUENCY statement, which allowed the programmer to give the probabilities associated with the paths followed after an IF statement. This facility allowed some optimisation of the use of index registers. For some obscure reason, FREQUENCY did not survive into later versions of FORTRAN.

A possible extension concerned entirely with user convenience is more flexibility about numerical constants at input. We suggest that input routines should accept whatever number of digits is offered and arrange the conversion to internal form so as to ignore (truncate or round) insignificant digits, rather than signal a fault. It is annoying and baffling to the average user to discover that the single precision value of π is 3·141593 on one machine and 3·1415926535 on another and that the first machine refuses to accept the accurate version, even if it cannot store it. It is also a totally unnecessary restriction that can easily be removed.

Conclusion

It has always been the claim of FORTRAN supporters, when arguing for the merits of the language against the virtues of its competitors, that it makes efficient use of a computer. It is in this spirit that we have drawn attention to several areas where agreed extensions are likely to be of value in the modern machine environment. Several of our suggestions presuppose particular forms of hardware facilities: compilers on more restricted machines should simply ignore statements intended to make better use of sophisticated equipment, rather than throw out the program as 'containing errors'.

We know that many of our proposals have, in fact, been implemented in some form or another on a variety of machines. Often, however, these extensions have been done in incompatible ways or else they have remained purely local. It is our hope that some action will be taken to ensure that those language extensions that are thought to be generally desirable are made in an agreed manner. Only if this is done will FORTRAN remain a viable language in general use among a wide and varied body of users.

We are grateful to Mr. M. J. R. Healy for permission to use the title of his article and also for his helpful comments.

Reference

HEALY, M. J. R. (1968). Towards FORTRAN VI? *Comp. J.*, Vol. 11, pp. 169–172.

Book Review

Mathematical Theory of Switching Circuits and Automata, by Sze-Tzen Hu, 1968; 253 pages. (University of California Press, $9.)

The development of switching theory over three decades has reached a point at which this book is particularly welcome. As Dr. Hu writes in his preface, most of the major problems have now been solved, and it is time to organise these results as a branch of pure mathematics in a way which will reveal their basic simplicity to both mathematician and engineer.

Rather than attempt to cover the entire field, the author prefers to take a few important topics and demonstrate the possibility and the value of carrying out such formalisation and simplification. The outcome is a structure of ideas which certainly have implications outside switching theory as such. A relationship is clearly visible, for instance, between the theory of prime implicants and that of 'resolvents' in logical calculi. Much of this book may be regarded as a valuable contribution to the study of finite Boolean Algebras.

After an introduction to switching functions and their various representatives, we are given a very thorough discussion of methods for finding the most economical disjunctive or conjunctive canonical forms for arbitrarily given functions. Here the terminology follows the topological approach of Mueller and Roth. This leads us to the more general problem of realising functions with a given set of logical devices. The notion of decomposition of a function in terms of simple decompositions develops the ideas of R. L. Ashenhurst, and this section culminates in a minimisation algorithm, described and illustrated with typical care.

In the last chapter the reader may perhaps consider himself to be taken on an unnecessary detour. Dr. Hu had decided to deduce the properties of finite-state machines as special cases by first setting up a theory of sequential machines with possibly infinite sets of states. If this is the hardest part of the book to assimilate, it is not unnecessarily so, once one accepts the worthwhileness of the approach; the author's explanations are always painstakingly clear. An automaton is defined as a function from the set of all input tapes into [0, 1], an n-input sequential machine as a function from $S \times [0, 1]^n$ into S, where S is a possibly infinite state-set. That every automaton is realisable by some sequential machine is almost trivial. That for each automaton there exists a (finite or infinite) 'minimal' machine which realises it is a result of some depth, and it receives a notably elegant treatment from Dr. Hu.

One might hesitate to recommend this volume to anyone completely unfamiliar with switching functions, but to those with at least slight experience of the practical problems involved and consequent motivations it should prove highly stimulating. A set of well chosen exercises consists in part of straightforward applications of the text, in part of developments of the theory that tie up with references in the deliberately concise bibliography. The book is pleasantly produced, and printing errors have been minimised.

MICHAEL BELL (London)