Permutations by interchanges

By B. R. Heap

Methods for obtaining all possible permutations of a number of objects, in which each permutation differs from its predecessor only by the interchange of two of the objects, are discussed. Details of two programs which produce these permutations are given, one allowing a specified position to be filled by each of the objects in a predetermined order, the other needing the minimum of storage space in a computer.

In programs of a combinatorial nature, it is often required to produce all possible permutations of N objects. Many methods can be used for this purpose and a general review of them has been given by D. H. Lehmer in *Proceedings of Symposia in Applied Mathematics* (American Mathematical Society), Vol. 10, p. 179. In this note we shall describe methods for obtaining the permutations in which each permutation is obtained from its predecessor by means of the interchange of two of the objects. Thus $(N-2)$ of the N objects are undisturbed in going from one permutation to the next.

We shall consider values of N up to $N = 12$, since the amount of time required for the generation of permutations for $N > 12$ is excessive. However, the methods are readily extendible should higher values be required.

We consider the problem of permuting the N letters $A, B, C, \ldots, Q$ amongst the N cells numbered $1, 2, 3, \ldots, N$, by specifying $(N! - 1)$ pairs of integers, each pair giving the numbers of the cells whose contents have to be interchanged in order to proceed to the next permutation. For small values of N the various pairs can be listed straightforwardly, and this is probably the fastest method if space is available. However, for larger values of N this is out of the question and a systematic method of obtaining the pairs of numbers is required. To do this, consider an inductive method for the permutations of n objects. Let us assume that a list of interchanges can be found which permutes $(n-1)$ objects. If we now fix the object in the nth cell, we can permute the first $(n-1)$ objects amongst the first $(n-1)$ cells. Now interchange the nth object with one of the first $(n-1)$ objects and again permute the first $(n-1)$ objects. Again interchange the nth object with one of the first $(n-1)$ objects, making sure that this object has not previously occupied the nth cell. Now repeat the process until each of the objects has filled the nth position while the other $(n-1)$ have been permuted, and clearly all $n!$ permutations have been found. Finally, it is clear that two objects can be permuted by a simple interchange, and so N objects can be so permuted. To achieve this one only needs to specify a total of

$$1 + 2 + 3 + \ldots + (N-1) = \tfrac{1}{2} N(N-1)$$

pairs of numbers. The actual specifying of the pairs is a matter for the individual as there are many ways of doing this. As an example the scheme in Table 1 used for N objects has the Nth cell successively occupied by $Q, \ldots, D, C, B, A$. (The original configuration is assumed to have A in cell 1, B in cell 2, etc.)

A program for computing the complete list of interchanges (and thus all permutations) from a skeleton list such as the one in Table 1 is fairly straightforward. For this purpose, we divide the triangular array in the table into $(N-1)$ lists $B_2, B_3, \ldots, B_N$, corresponding to the columns of the array, so that B_M contains $(M-1)$ entries. Only the first number of each pair need be stored, as the second number will be available from a register M. Integer registers $A_2, A_3, \ldots, A_N$ are also required for use in picking out the entries in the lists. The flow diagram is then straightforward and is given in Fig. 1, the pair of numbers denoting each interchange being contained in X and Y.

Table 1

Skeleton lists of interchanges, giving the order of occupation of the Nth cell Q, P, . . ., D, C, B, A

$N =$	12	11	10	9	8	7	6	5	4	3	2
Interchanges	9,12	9,11	7,10	7,9	5,8	5,7	3,6	3,5	1,4	1,3	1,2
	6,12	7,11	8,10	1,9	2,8	3,7	4,6	1,5	2,4	1,3	
	3,12	5,11	1,10	5,9	7,8	1,7	3,6	3,5	3,4		
	10,12	3,11	6,10	5,9	2,8	5,7	2,6	1,5			
	9,12	1,11	5,10	3,9	1,8	3,7	3,6				
	4,12	9,11	4,10	3,9	2,8	1,7					
	3,12	7,11	9,10	7,9	3,8						
	8,12	5,11	2,10	1,9							
	9,12	3,11	3,10								
	2,12	1,11									
	3,12										

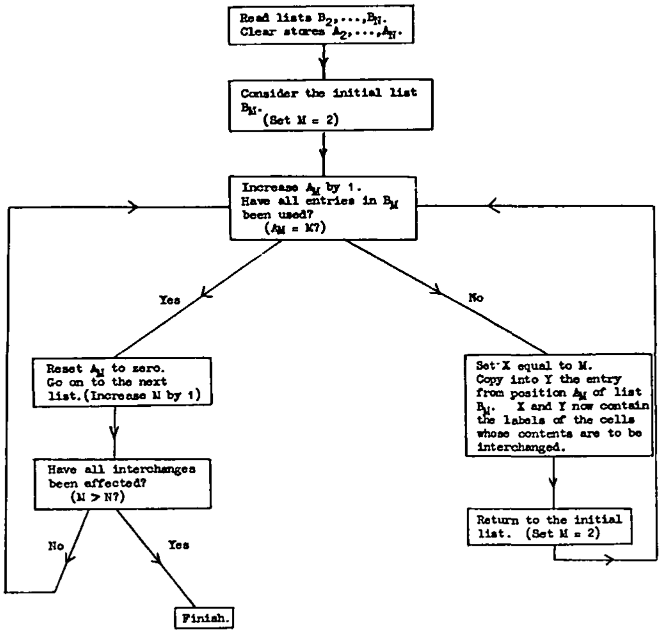

Fig. 1.—Flow diagram of program for extracting the complete list of interchanges from a skeleton list similar to that of Table 1

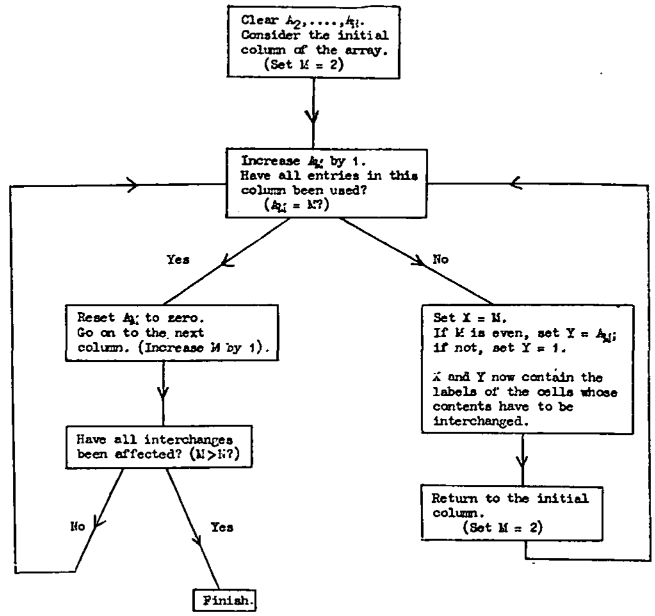

Fig. 2.—Flow diagram of program for obtaining the complete list of interchanges for the permutations of *N* objects

Whereas the above method is useful in that the objects can be made to occupy the Nth cell in a predetermined order, it is necessary to store the skeleton list of interchanges. If a special order of occupation of the Nth cell is not required, or if storage space is short, an alternative program can be used, in which none of the space previously occupied by the lists B_M is needed. The program uses the same general method as the one given above; i.e. for n objects, first permute the first $(n-1)$ objects and then interchange the contents of the nth cell with those of a specified cell. In this method this specified cell is always the first cell if n is odd, but if n is even, it is numbered consecutively from 1 to $(n-1)$. For example, the list of interchanges for five objects begins:

1,2 1,3 1,2 1,3 1,2 1,4 1,2 1,3 1,2 1,3 1,2 2,4

1,2 1,3 1,2 1,3 1,2 3,4 1,2 1,3 1,2 1,3 1,2

These 23 interchanges permute the first four objects and in fact are the same as those given by the list of Table 1. However, the next interchange is 1,5 instead of 3,5 as in the table. The above list is then repeated, again followed by a 1,5 interchange, and the process is repeated until all 119 interchanges have been effected.

That this scheme produces all possible permutations is obvious on inspection for the first few values of N. The flow diagram of Fig. 1 needs little amending for this program, and the new flow diagram is shown in Fig. 2.

The objects now do not occupy the Nth cell consecutively, and in Table 2 we give the order of occupation of the Nth cell for values of N up to 12.

Table 2

Order of occupation of the *N*th cell when the flow diagram of Fig. 2 is used

$N=$	2	*B*	*A*										
	3	*C*	*B*	*A*									
	4	*D*	*C*	*B*	*A*								
	5	*E*	*B*	*C*	*D*	*A*							
	6	*F*	*E*	*B*	*C*	*D*	*A*						
	7	*G*	*D*	*C*	*B*	*E*	*F*	*A*					
	8	*H*	*G*	*B*	*C*	*D*	*E*	*F*	*A*				
	9	*I*	*F*	*E*	*D*	*C*	*B*	*G*	*H*	*A*			
	10	*J*	*I*	*B*	*C*	*D*	*E*	*F*	*G*	*H*	*A*		
	11	*K*	*H*	*G*	*F*	*E*	*D*	*C*	*B*	*I*	*J*	*A*	
	12	*L*	*K*	*B*	*C*	*D*	*E*	*F*	*G*	*H*	*I*	*J*	*A*

The above schemes can readily be extended to cases where the N objects fall into distinct groups and it is required to perform all permutations of the objects within these groups. In this form the author has used the method for running over all possible labellings of the points of unlabelled graphs in order to calculate certain properties of these graphs. It would seem that programs of the above type are especially useful for this purpose owing to the minimum disturbance between successi e permutations.

I am grateful to Mr. G. G. Alway and Dr. J. L. Martin for discussion. The work described in this paper forms part of the research programme of the National Physical Laboratory and is published by permission of the Director of the Laboratory.

Large-Capacity Memory Techniques for Computing Systems. Edited by Marshall C. Yovits, 1962; 440 pages. (New York: *The Macmillan Company* $11.00. London: *Macmillan Company New York*, 75s.)

This book consists of an edited version of the papers presented at the Symposium of the same name, which was held in Washington D.C. May 23–25 1961 and sponsored by the Information Systems Branch of the Office of Naval Research.

The object of the Symposium, it was claimed, was to focus attention on new ideas, research and development which could lead to novel computer memories capable of storing very large amounts of information. The 30 papers, which are now collected within 440 pages, were all invited by a committee, under the Chairmanship of Marshall C. Yovits, that was set up for this purpose.

In the preface to the book Marshall C. Yovits has made it clear that the term "large capacity" has deliberately been left vague, on the grounds that the techniques rather than the operational capability are of interest. However, it is stated that it is hoped that stores with a capacity of hundreds of millions of bits if not billions of bits will eventually result. Furthermore these should have access times commensurate with the fastest computing system now on the horizon.

The thing that struck me when taking a first glance at the book, was the sequence in which the papers were arranged. I appreciate the difficulties with the number of overlapping techniques but I can see no reason for the only two papers on cryogenic memories to be placed No. 20 and No. 30. I think that in addition to the normal contents list a simple cross-reference table would have been a great help in finding one's way around the various techniques.

The first two papers deal with what are probably the two most important topics of any storage system, i.e. access techniques and organization. The first of these two papers I found disappointing and I passed rapidly to the second. This paper, by Ledley, dealt with the organization of large-capacity memory systems, and was reporting work supported by the Symposium organizers, and it so happened that this was the longest paper in the collection. Nevertheless the paper does a very useful job of work in covering simple, complex and indirect addressing, also list co-ordinate and both simple and complex contenting. 36 references are included but, as is pointed out, this is rather misleading, for literature on the organization of stores is sparse. However, I think the author has done his best to improve this situation. In conjunction with this contribution it is worth reading the paper by Goldberg and Green which covers some of the practical approaches to stores whose items are retrieved on a basis of content rather than location.

As may be expected at a symposium of this type, photographic storage was well to the fore. The Bell semi-permanent store is described in a paper by Hoover and Haugk, and there are three papers on optical selection systems for photographic stores. Brian and Focht of Philco go into details of the problem of positioning the spot of a C.R.T. so as to locate the required address accurately. The other two papers are based on a continuously rotating mirror to scan a photographic plate, and the use of a matrix-controlled electroluminescent screen to position the light spot. A completely new approach to photo storage is given in a paper "The Photochromic Microimage Memory" by staff from the National Cash Register Co. A photochromic coating is used to replace the normal photographic emulsion. This photochromic coating consists of a molecular dispersion of reversible light-sensitive dyes that produce a high resolution grain-free material. A reduction capability of 200/1 is claimed and, as is pointed out, with this reduction the contents of 40,000 filing cabinets can theoretically be reduced to a single cabinet.

Two papers on large-capacity document storage are included; these cover the Magnaview and the IBM systems. In the case of Magnaview, both photo and magnetic storage are combined.

There are six papers on various aspects of magnetic recording. A paper by Angel describes the N.C.R. M.C.R.A.M. (Magnetic Card Random Access Memory) which is in fact now in full production. There is a group of three papers from IBM covering three aspects of magnetic disc recording. The first covers the problems involved in the preparation of the disc and the magnetic material, the second covers transducers and amplifiers, and the third covers the air-lubrication slider bearing for controlling the space between the transducers and the magnetic material. The other two papers on magnetic recording deal with novel ways of reading and recording. The paper on reading is based on the use of the Kerr magneto-optical effect for detecting the direction of magnetization along the preferred axis of small areas of a continuous thin magnetic film, which can be deposited on either a drum or a disc. There is only a summary of the paper on a novel method of recording. The proposal is to use an electron beam to raise a small area of a magnetic film beyond its Curie point. If the film was pre-magnetized normal to its surface then, when the small heated area is cooled, it will reverse its direction of magnetization. This is the necessary consequence of the fact that minimum magnetic energy requires flux closures through the temporarily non-magnetic area.

In addition to the above there are four other papers based on the use of thin magnetic films; one of these, by Fuller and Robinstein, considers the high density of magnetic domains that are possible in thin magnetic films, and goes on to consider ways that this may be utilized for large capacity files. A paper by Meier and Kolk of N.C.R. describes the fabrication and use of a cylindrical thin magnetic film memory element, and Howard *et al.* report on the results of their investigation of woven-screen memory techniques. Finally a paper by Bittmann covers the more conventional flat plate type of thin magnetic film store.

There is only one paper based on the use of ferrite, and this is in the form of long hollow extended cylinders. These are aligned and subsequently cut to form ring cores. A density of 50 per inch is claimed.

There are three papers, in addition to those on photo storage, on changeable permanent stores. Two of these are by Japanese authors and both use the presence or absence of holes in a metal plate to represent "1" and "0". The difference between the papers is that in one case the presence or absence of holes is sensed by the variation of inductance, and in the other by the difference in capacity. The third paper on this topic uses small permanent magnets arranged on a card to represent the binary digits, and the presence or absence of these is sensed by the switching of a nearby "twistor".

The book contains papers making the cases for ferroelectric, phosphor and electron spin echo phenomena, but none of these gives the impression that they will get off the ground. Of the two papers on cryogenic stores the first is little more than a series of design specifications, without any details as to how they were determined. However, the second, and the last in the book, is a very good paper by Burns and his colleagues from R.C.A., giving a clear account of their particular storage system.

Reviewing this book more than two years after the Symposium has made me query whether it is time that someone did something to speed up the publication of information of this nature; for it loses so much of its impact if it fails to have rapid dissemination. Would, in fact, a simple paperback edition have speeded up the process? Nevertheless the problem of large capacity storage is, as far as I know, a long way from a satisfactory solution, and it is always useful for those interested in using or designing this type of store to have at their disposal a collection of papers that show what other workers in the field have done in the past.

G. H. PERRY.

Digital Techniques. By D. W. DAVIES, 1963; 158 pages. (London: *Blackie & Sons Ltd*. 30s.)

This book provides an introduction to many of the methods used to handle digital information which, although originally developed for digital computers, are becoming increasingly used in other applications such as data transmission, machine tool control, and instrumentation.

Chapter 1 gives a brief account of the essentials of a digital system, including methods of gathering data, and the magnitude of errors. Chapter 2 discusses the collection of data in binary and decimal form, the use of code plates, code discs, optical and magnetic digitizers, a simple analogue to digital convertor, and the need to use cyclic codes or redundant tracks to avoid reading ambiguities. The rule for translating cyclic binary to normal binary code is given, but without logical equations or a block diagram of a system which could perform the operation.

Chapter 3 deals with the design of "and" "or" and "inverting" gates, and bistable and counting circuits, and illustrates their performance by truth tables. Their practical realization using transistors, diodes and thermionic valves is described in Chapter 4. Current-steering or long-tailed pair circuits using valves and transistors, direct-coupled transistor logic, non-saturating transistor circuits, and tunnel-diode/resistor logic are mentioned briefly. The treatment of saturated transistor circuits on p. 52 may mislead by including the relation $i_c = \beta i_b$, and could be made more instructive by including a diagram of the collector characteristics of the transistor, showing the load line, and the linear, cutoff and saturation regions. Again on p. 59, the discussion of direct-coupled logic could be condensed if a diagram superimposing the collector and input characteristics of a suitable transistor were given, and in Fig. 4.23 a much simpler "and" gate can be constructed by merely connecting two transistors in series, each base being connected to one of the inputs.

Chapter 5 describes circuits which include reactive elements, such as monostable multivibrators, delay lines, blocking oscillators, square-loop magnetic core circuits and core-transistor circuits. A simple description of a coincident-current core store is given, but there is no mention of recent developments such as partial flux switching and other methods of selection. On p. 76 the common-base current gain of a transistor is designated as β instead of α, and on p. 75 the field and flux equations could be presented in a simpler form by using the M.K.S. system of units.

Binary counting circuits using valves and transistors, and methods of connecting them to produce decade counters are described in Chapter 6. Although three methods are described, these all require a delay element, and circuits which include only logical inter-connections, which are preferred for high-speed decade counters, are not mentioned. The alternative decimal gas-filled and beam-deflection counters are described briefly, but no mention is made of "auto-transfer" operation, or single-pulse gas-filled tubes, and their maximum counting speed is now 1 Mc/s, not 100 Kc/s as quoted on p. 98.

The use of punched cards, punched paper tape, and magnetic tape for the long-term storage of data and for changing its transmission rate is described in Chapter 7, which includes photographs and descriptions of paper-tape readers and magnetic-tape decks, and some discussion of RZ, NRZ and phase modulation schemes for digital magnetic recording. The clear advantage of the phase modulation system when transformer-coupled heads are used is, however, omitted.

Chapter 8 is concerned with the various printing and display devices such as lamp and gas-filled indicators, xerographic and matrix printers, electric typewriters, and X-Y plotting tables. The general principles of machine-tool control by digital signals are also mentioned briefly.

The final chapter mentions some topics in system design, including the selection of the number of digits to be encoded, their rate of transmission and some of the logical problems which may be encountered. These are illustrated in relation to a bi-directional counting system, and a digital magnetic-tape unit.

The treatment is generally expository rather than analytical, and few of the circuit diagrams give component values, so that the book will disappoint students interested in the design of digital circuits. A particular fault is the failure to discuss operating speed, which is an important factor in any digital system. Bearing in mind the present state of the art, it might be better to omit all circuits involving thermionic valves in order to devote more space to transistor circuits, and their response times.

It is also most surprising to find in a book on digital techniques no mention of Boolean algebra as a means of describing and manipulating logical operations. Any student of digital systems will encounter this notation in nearly all textbooks and periodicals, and an introduction to it would surely be more useful than, for example, the description of lamp display devices, or the mechanism of high-speed printers.

Many manufacturers now produce standard logical blocks which may be inter-connected to form large and complex digital systems with facilities for producing high power outputs to operate tape punches, fluid valves, etc., and which may easily be modified or extended. These are now widely used in industrial control systems and one would expect to find them mentioned.

In view of the omissions mentioned above, the book could be recommended only as an introductory text for electrical engineering students of degree level, to be followed by a fuller treatment of logical algebra and the design of digital circuits. It would, however, be suitable for less academic courses, or for civil, mechanical or chemical engineers who encounter digital instrumentation and wish to obtain some knowledge of the principles involved.

J. C. CLULEY.

Theory and Design of Digital Machines, by THOMAS C. BARTEE, IRWIN L. LEBOW, IRVING S. REED. 1962; 324 pages. (London: *McGraw-Hill Publishing Co. Ltd.*, £4. 9s. 0d.)

Many different disciplines are involved in the design of digital computers and "an unlikely mixture of engineers, logicians, mathematicians and philosophers have contributed to the present-day state of the digital art."

Each regards the computer in the light of his own specialization. Thus, the engineer views the design problem as one of circuit design and of interconnecting systems components of different types to provide the required systems function. In this he tends, sometimes, to miss the subtleties of a more abstract mathematical approach although often creating ingenious solutions, bordering on genius, in the process.

On the other hand, the abstract approach of the mathematician, or logician can frequently fail to appreciate a lucrative area for investigation, which a familiarity with practical design problems would show.

The problem is basically one of communications between the many groups engaged in computer research. The overwhelming need is for a common language, which would enrich the activities of each group, facilitate the integration of the many approaches to computer design and lead to a much-needed broadening of the overall theory.

The book under review makes some progress towards this goal. Aimed at the practising engineer and student it combines the more traditional specializations of machine design and switching theory, and presents an integrated, systematic mathematical approach to computer design, based on a symbolic representation of transfer operations between registers.

Thus, a digital computer unit is regarded as a set of independent and dependent registers interacting through transfers specified by a control unit. A realistic computing system is then treated as a set of interacting computing units.

The design process is considered in three distinct phases:

(*a*) systems design, which sketches in the general configurations of the machine and specifies the general class of hardware to be used;

(*b*) structural design, which describes the system in terms of transfer relations;

(*c*) logic design, which realizes the transfer relations by means of Boolean equations.

Accordingly the book divides naturally into three basic sections:

(*a*) the foundations of machine design, in which the basic components, such as the register and combinational network, are introduced and defined, together with a mathematical notation which conveniently describes their operation;

(*b*) a theoretical treatment of the mathematical foundations of switching theory based on Boolean rings and fields, and systematic methods of minimizing Boolean expressions using the well-known Quine-McCluskey method and introducing more recent algorithms;

(*c*) various aspects of machine design, introducing the basic sequential networks and showing how, with the addition of a simple control unit, an elementary computer can be evolved. This concept is expanded through the addition of the arithmetic unit and a program unit to a simple general-purpose computer. The full design method is then developed and illustrated with two special-purpose computer designs, the first an automatic radar detection and processing computer and the second a digital differential analyser.

A final chapter discusses more advanced, theoretical subjects such as minimal-state machines, Turing machines, probabilistic machines and finite-state automata. Number representation systems are dealt with in an Appendix.

The treatment is limited to logical design subjects and makes only passing reference to programming, electronic-circuit designs and numerical methods.

The book is very well written in a lucid style although, at times, rather simple concepts are unnecessarily complicated in the interests of mathematical rigour. Questions follow each chapter to illustrate and amplify the material in the text. Copious references are provided after each chapter for further study, although there seems little point in the needless repetition this involves.

The mathematical rigour of the treatment makes heavy going for the engineer familiar with more direct methods of design, and the authors do not always show convincing reason why the rigour is necessary. Thus the Quine-McCluskey technique is presented as a manual method for switching-circuit minimization, whereas its elegance and popularity is largely due to the ease with which it can be programmed for a computer-based design procedure. It is the general adoption of computer techniques for computer design that makes mathematical methods such as those presented by the authors so important, and makes the book itself worthwhile reading for all interested in computer design.

K. L. SMITH

A Survey of Mathematical Logic, by HAO WANG, 1963; 651 pages. (Amsterdam: *North-Holland Publishing Company*, 120s.)

Although this is primarily a collection of papers, it is more than just a collection, for some of the papers have been amplified and the collection welded into a whole by the introduction of supplementary matter. Wang writes clearly and persuasively and has interesting and important things to say about a wide range of topics.

The book is divided into five parts. The first of these starts with a chapter on the axiomatic method, and contains informal accounts of Gödel's famous incompleteness theorems. This is followed by a reprint of the paper, "Eighty years of foundational studies," which first appeared in the Festschrift of Bernays (*Dialectica* 1958), and a paper on the axiomatization of arithmetic (from the *Journal of Symbolic Logic*, 1957). The section concludes with an excellent discussion of the concept of computability, which contains an account of Mucnik and Friedberg's solution of Post's problem in which they showed that there is a recursively enumerable set B (i.e. a recursive function f) which is not recursive (i.e. $Ex\{y = f(x)\}$ is not a recursive relation), and a recursively enumerable set A which is not recursive in B.

The second part of the book is devoted to the theory of calculating machines and reproduces papers from the *Journal of the Association for Computing Machinery*, *the Zeitschrift f. math. Logik*, and the *IBM Journal of Research and Development*, on universal machines, the logic of automation, the proof by a machine of 400 theorems in *Principia Mathematica*, and circuit synthesis by solving sequential Boolean equations (i.e. Boolean equations with an additional "time" difference operator).

The third part opens with a general discussion of predicate logic, of the first and higher orders, and the arithmetization of metamathematics, with emphasis on the consistency problem, and Gödel's theorems. Skolem's non-standard model of arithmetic (a set of functions of ordinal greater than ω which play the part of the natural numbers in a formalized arithmetic), and Ackermann's consistency proof by transfinite induction are clearly explained.

In the fourth and fifth parts on impredicative and predicative set theories we move into the field in which Wang has done his best-known work. The fourth part opens with a very informative survey of a variety of axiomatizations of set theory, the Zermelo theory, that of Paul Bernays, Russell's theory, those of Quine, and Gödel's set theory. This is followed by a reprint of a paper from the *Mathematische Annalen* (1953) on relationships between number theory and set theory, and a summary of the contents of Wang's Harvard (1948) doctoral dissertation, much of which has been published in the *Journal of Symbolic Logic*; another paper from that Journal, on the formalization of mathematics, discusses the problems of consistency and adequacy. The book ends with a paper from the *Zeitschrift f. math. Logik* (1959) on ordinal numbers and predicative set theory, which discusses Herbrand arithmetic definitions and Herbrand partial recursive definitions.

Published simultaneously by the North Holland Publishing Company and The Science Press, Peking, this book was printed and bound in China. The printing is good and clear, but the paper is not of the quality to which the Dutch printers have accustomed us in this series. The quality of the contents is happily beyond question.

R. L. GOODSTEIN

THE COMPUTER JOURNAL

Published Quarterly by

The British Computer Society, Finsbury Court, Finsbury Pavement, LONDON, E.C.2, England.

All inquiries should be sent to the Assistant Secretary at the above address.