133[F].—F. L. MIKSA, *Table of primitive Pythagorean triangles with their perimeters arranged in ascending order from* 119992 *to* 499998. 506 typewritten leaves on deposit in UMT FILE.

This table is a continuation of UMT 111 [*MTAC*, v. 5, p. 28], a table by A. S. ANEMA to 120000. The introduction gives a table showing the number of triangles whose perimeters do not exceed P for $P = 120000$ (10000) 500000. The total number of these triangles is 35114. LEHMER'S asymptotic formula gives 35115. Similar data are given for pairs of triangles having equal perimeters of which there are 1750. There are 65 cases of 3, and one case of 4 isoperimetric triangles.

F. L. MIKSA

613 Spring Street
Aurora, Illinois

134[L].—Y. L. LUKE & D. UFFORD, *Tables of* $\int_0^\infty (zt)^{-1}(z + t - \sqrt{z^2 + t^2})e^{-it}dt$. 8 mimeographed leaves on deposit in UMT FILE and also available from Midwest Research Institute, Kansas City 2, Missouri.

The tables give the real and imaginary parts $U + iV$ of the integral given in the title together with the function

$$U + \log 2z + \gamma - 1.$$

6D values of the three functions are given for $z = 0(.01).1(.1)4(.2)5$.

135[L].—J. E. WILKINS JR. & NINA KROPOFF, *Table of Laguerre Functions*. Seven mimeographed leaves on deposit in the UMT FILE.

The table gives 4D values of

$$L_n(x)/n! = M(-n, 1, x) = \sum_{k=0}^{n} (-x)^k \binom{n}{k}/k!$$

for $n = 2(1)7$ and $x = 0(.1)10(.2)20$. [See *MTAC*, v. 1, p. 361, 425, v. 2, p. 31, 267.]

J. E. WILKINS, JR.

Nuclear Development Associates
80 Grand Street
White Plains, N. Y.

AUTOMATIC COMPUTING MACHINERY

Edited by the Staff of the Machine Development Laboratory of the National Bureau of Standards. Correspondence regarding the Section should be directed to Dr. E. W. CANNON, 415 South Building, National Bureau of Standards, Washington 25, D. C.

TECHNICAL DEVELOPMENTS

Provision for Expansion in the SEAC

In developing the SEAC, two divergent objectives had to be attained. The first objective was to get a modest performance high-speed computer into operation at the earliest possible date; the second objective was to

develop an effective experimental tool for evaluating advanced computer components and systems. In order to reconcile these conflicting aims, it was decided that the SEAC should be made *expandable*, i.e., should consist of a central nucleus (with a high priority on early completion) to which additional units could be added later, as time permitted or as initial operating experience suggested.

The central nucleus of the SEAC was completed and put into regular operation in the Spring of 1950. Since then, several additional units have been incorporated into the machine's memory, input-output, and control equipment.

1. *Memory.*—The first major change made in the SEAC was to double the size of the high-speed internal memory. Initially the SEAC was operated with an acoustic delay-line memory holding 512 words stored serially in 64 mercury tanks (with average access time: 168 microseconds). Subsequently an additional high-speed memory was incorporated which holds 512 words stored in 45 Williams' tubes. All 45 tubes simultaneously supply the computer with 45 binary digits in parallel (with average access time: 12 microseconds). Although the SEAC is in other respects a serial machine, provision was made in the original design for the use of both serial and parallel types of memory. Conversion of the word-form back and forth from serial to parallel is accomplished by means of a shift register with a special flexible control.

2. *Input-Output.*—The same shift register equipment is sufficiently versatile to allow the SEAC to operate with completely asynchronous, serial or parallel, input-output equipment. For example, the SEAC has been operated in conjunction with Teletype punched-paper tape, with Teletype keyboard and printer, with a visual (cathode-ray tube) high-speed graph plotter, with single-channel magnetic wire, and with single-channel magnetic tapes. Additional units which are planned to be incorporated in the near future include multi-channel magnetic tapes, Flexowriter punched-paper tape and printing equipment, punched-card-to-magnetic tape conversion equipment, and a high-speed photographic printer. Also, an external selector is being added to the SEAC which will be capable of selecting (under the computer's control) from among any of ten different input-output units.

3. *Control.*—Another major change to be made in the SEAC will be the addition of equipment to the control. Provision was made for this expansion in the original design of the machine, and, as a result, the operator of the computer will be able to choose (before entering his problem) between two distinct modes of operation (i.e., between the 4-address or the 3-address mode) merely by throwing a few toggle switches.

The main distinction between the 4-address and the 3-address mode is indicated in Table I, which lists the information contained in an instruction-word in each system. Note that the first three items of information are the same in both systems. These items indicate where in the memory the operands are located (α and β), and also where in the memory the result of a given operation on them is to be sent (γ). In the 4-address system, the instruction also contains an explicit statement on where the *next* instruction is located in the memory (δ). In the 3-address system, however, there is no such explicit statement. Instead, instructions in this system are automati-

cally sequenced, i.e., successive instructions being located, ordinarily, in successively-numbered memory positions.

The types of operations which can be performed in both the 3-address and 4-address systems, are similar. They include (besides input-output operations) addition, subtraction, multiplication, division, and a type of extract operation called "logical transfer." Also, three types of comparison operations (sometimes called discrimination or branch operations) can be performed. In a typical comparison operation, when the result of the comparison between two operands indicates that the first operand is greater than or equal to the second operand, then the next instruction is chosen normally, in the manner just described. *But*, if the second operand is the greater, then the next instruction in both systems is chosen from address γ.

4. *Floating Address Feature.*—In the 3-address system, a special automatic feature is included which is believed to be novel and which is referred to as the Floating Address feature. This was incorporated into SEAC in order to shorten and simplify the work of the programmer. It does not accomplish any effects which the human programmer could not also accomplish at the cost of additional labor. The scheme does, however, automatize certain time-consuming programming operations, thereby completely relieving the programmer of any need for concerning himself with them. This is accomplished at the cost of a relatively small amount of additional equipment.

In brief, this feature makes it easy for the programmer to reutilize standard routines or lists of instructions which have already been compiled for carrying out certain frequently-used operations, such as taking a square root, making a binary-to-decimal conversion, and so on. Obviously he would like to make full use of any such material originally prepared for one problem in the solution of other problems. But there is a difficulty in the way of such straightforward re-use of standard routines. This arises from the fact that the *memory locations* into which he must insert the standard list of instructions will be different in general for each new problem. This difference forces him to change the memory address by which he identifies each individual instruction in the routine each time he uses the routine in a different problem. The *floating address* feature performs this change of address *automatically* each time the routine is used, and in a manner which requires no attention on the part of the programmer.

Table II illustrates the characteristics of the floating address feature. In preparing a routine originally, the programmer can designate any address as either *absolute* or *relative*. An absolute address is interpreted in the usual sense merely as a number identifying a specific memory location. A relative address, however, identifies a position in the memory by specifying its displacement relative to the position of the instruction-word itself. To illustrate how this works: if an instruction located in memory position m specifies that the address of the first operand is number 17 *relative*, then the control will take the word in memory location $m + 17$ and use it as the first operand. To refer to location $m - 17$, the programmer uses as address the complement of 17 ($2^{12} - 17$, relative). It is possible to use any combination of absolute or relative addresses in any instruction word.

By utilizing this feature, therefore (by using relative-address designations for all references to instructions inside the routine itself), routines may be prepared in advance once and for all, transcribed on tape, and

stored away in the library files. For any subsequent problem, then, each routine can be inserted in the memory in any arbitrary location whatever, and executed immediately without need for any preparatory modification to adapt it to the particular memory location in which it happens to fall.

The second major characteristic of the floating address feature is intended to relieve the programmer of another related type of burden. To illustrate this, let us represent the various storage locations of the internal memory by the boxes shown on the flow chart (Table III). Each box contains an instruction. Each group of boxes represents a typical routine, a list of instructions stored in consecutively numbered address locations. The right-hand list represents a sub-routine describing some standard process which has been entered into the memory verbatim from a strip of library tape. The left-hand list represents the main routine prepared specially for the problem at hand. During the course of any problem, the control usually follows the main routine for a while, then jumps to some sub-routine, then jumps back to the main routine, and so forth. Now the same sub-routine generally will be referred to repeatedly during the course of a problem. (E.g., square roots may be sprinkled in many places throughout the computation.) The point of departure from the main routine will in general be different in each case. Under the 4-address SEAC system, the programmer has to arrange, before embarking on the sub-routine, to plant some sort of clue in some pre-arranged location (or at the end of the sub-routine itself) which will lead the control back to the main routine at the proper point of return each time. This means programming extra operations.

With the 3-address floating address system, the return to the proper point in the main routine always takes place automatically and requires no attention at all from the programmer. This is accomplished in the following manner: The system provides two small storage registers, each of which is capable of holding an address number. Normally, as soon as any instruction is executed, the contents of *one* of these registers are advanced one unit, and the number it contains is taken as the address of the next instruction. The contents of the other register are held unchanged. A single binary digit (called "d" on Table II) is reserved in each instruction word for designating which one of the two registers is to be referred to.

To be more specific, on main routines the instructions are normally written with the register designation digit $d = 0$. These main routines are sequenced by the counter register called #0. To embark on a sub-routine, the programmer writes a special "jump" instruction (with $d = 1$) which causes the other counter register (called #1) to take over. The sub-routine, always written with the digit $d = 1$, continues using counter #1. The *final* instruction of the sub-routine, however, is written with digit $d = 0$. Immediately after executing the final instruction, therefore, counter #0 is referred to for the address of the next instruction, and the control is back on the main routine at the point immediately following where it left off. All sub-routines are written in an invariant form in this respect, and no other attention on the part of the programmer is required.

Some of the alternative courses followed by the control are indicated in Table III: *normal* operation, *jump* operation, and the path of *automatic return*. By combining this dual-counter feature with the relative address feature, a number of other useful operations may be performed. As an

example of one: by a jump operation with a relative resetting for counter #0 at the end of the sub-routine, automatic return may be made to take place to any part of the main routine.

Now the question naturally arises: How much does this added control equipment increase the size of the computer? The answer in terms of the total tube count of the complete computer is: about 3% (30 tubes out of a total of about 1050). Since the SEAC was originally designed with the possibility of these additions in mind, the actual process of attaching new units is quite simple and straightforward.

Is annexation of such additional equipment justified? We believe it is. The value of the computer ought to be measured by its total productivity in terms of complete problems solved. Experience to date with the SEAC suggests that in the long run the main bottleneck on problem output may be the ability of the human operator to prepare and check adequately the programs that he enters into the machine. Preparing a program for a large scale problem is a highly intricate process and demands complete accuracy of detail. Furthermore, checking a program which appears to be going astray on the machine often turns out to be very time-consuming. It is found, for example, that the time necessary for locating and correcting *program* errors is often comparable with the time spent in locating and correcting *machine* errors. With the advent of (first) more complicated problems and (second) more reliable machines, the balance will become even more heavily weighted against the programmer. By simplifying his job and reducing the number of details with which he has to concern himself, we think we can reduce the incidence of human error and thereby gain markedly in efficiency. For this reason, we feel that the supplementary equipment just described will contribute substantially towards increasing the SEAC's problem-solving productivity.

ALAN L. LEINER

NBS Electronic Computers Lab.
Washington, D. C.

TABLE I. THE SEAC'S CONTROL

NOW IN OPERATION	*BEING INCORPORATED*
4-address system	3-address system (with Floating Address feature)
Each instruction word contains:	Each instruction word contains:
(1) Address of first operand (α) ... 10 binary digits	(1) Address of first operand (α) ... 12 binary digits
(2) Address of second operand (β) ... 10 binary digits	(2) Address of second operand (β) ... 12 binary digits
(3) Address of result (γ) ... 10 binary digits	(3) Address of result (γ) ... 12 binary digits
(4) Address of next instruction (δ) ... 10 binary digits	(4) Floating address information (a, b, c, d) ..... 4 binary digits
(5) Symbol specifying type of operation to be performed .... 4 binary digits	(5) Symbol specifying type of operation to be performed .... 4 binary digits
(6) Halt signal .... 1 binary digit	(6) Halt signal .... 1 binary digit
Total: 45 binary digits in each word	Total: 45 binary digits in each word

TABLE II. FLOATING ADDRESS INFORMATION

RELATIVE ADDRESS INFORMATION

(a) indicates whether α is an absolute or relative address .... 1 binary digit

(b) indicates whether β is an absolute or relative address .... 1 binary digit

(c) indicates whether γ is an absolute or relative address .... 1 binary digit

DUAL REGISTER INFORMATION

(d) modifies choice of next instruction ..... 1 binary digit

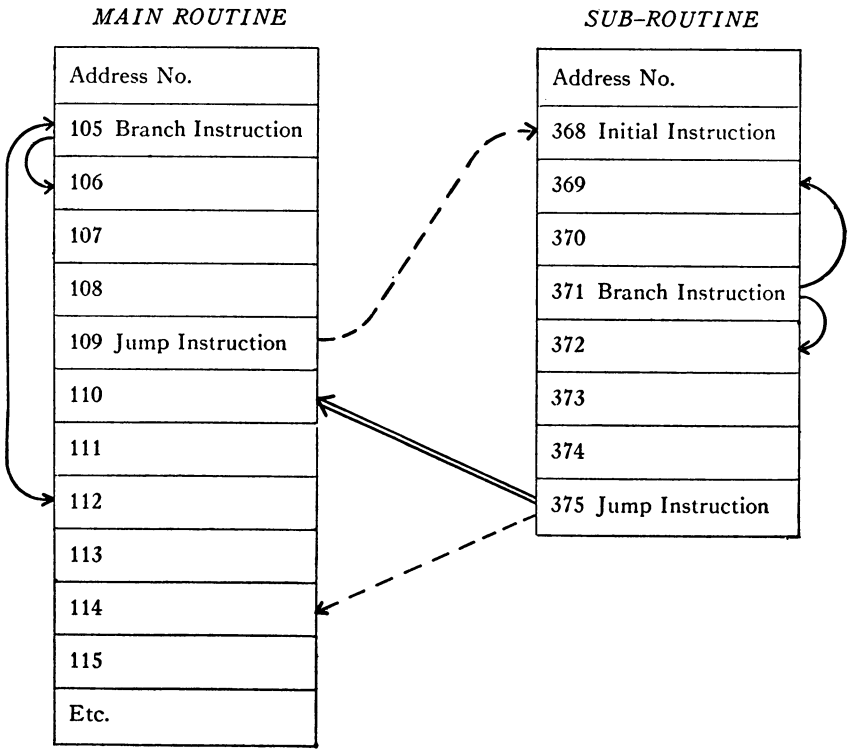
Total: ..... 4 binary digits

TABLE III. CHOICE OF NEXT INSTRUCTION IN THE THREE-ADDRESS (FLOATING) SYSTEM

POSSIBILITIES JUMP operation, indicated by: — — —→

NORMAL operation, indicated by: ⟶

PATH OF AUTOMATIC RETURN, indicated by: ⟹



Notes on Numerical Analysis—6

The Convergence of Seidel Iterants of Nearly Symmetric Matrices

If A is a matrix, we may write $A = D + L + R$. Here D is the matrix of its leading diagonal elements, L the triangular matrix of the elements

below this diagonal, and R the triangular matrix of the elements above. If B is the Seidel[1] iterant of A, then $B = -(D + L)^{-1}R$.

It is known that if A is symmetric, $\lim_{n\to\infty} B^n = 0$ if A is positive definite.[2] By continuity one may infer that if A is 'nearly equal' to some symmetric matrix C, the Seidel iterants of A and C would behave in a similar way with regard to convergence. For, although there is a limiting process involved, the convergence of B^n to 0 depends on the maximum modulus of the characteristic roots of B, and this is a continuous function of the elements of B. The problem attempted in this paper is to determine a precise meaning for the notion 'nearly equal.'

Any matrix A can be written in the form $A = C + S$, where C is symmetric and $S = \lambda - \lambda'$ is skew-symmetric (λ is defined in the Theorem). C may be regarded as the symmetric matrix nearest to A, and λ is then a measure of its asymmetry. Our solution of what is to be understood by 'nearly equal' is given in terms of C and λ. The result is simplest when $D = I$, the unit matrix. We state the result in that case as a theorem. The minor adjustments required when $D \neq I$, all $a_{ii} > 0$, are explained at the end of the paper.

Theorem. *If* $A = I + L + R$, $C = I + \frac{1}{2}(L + R') + \frac{1}{2}(L' + R)$, $B = -(I + L)^{-1}R$, $\lambda = \frac{1}{2}(L - R')$,

$$M = I + \lambda + \lambda' - \tfrac{1}{2}\{(\lambda - \lambda')(I + L + R)^{-1}(I + L - R) + (I + L' - R')(I + L' + R')^{-1}(\lambda' - \lambda)\},$$

then, provided M *is positive definite,* $\lim_{n\to\infty} B^n = 0$, *or* $\neq 0$ *according as* C *is or is not positive definite.*

For any matrix A, let $[N(A)]^2 = \Sigma_{i,j} a_{ij}^2$. If we write $M = I + M^*$, then it may be proved that M is certainly positive definite if $N(M^*) < 1$.

From known inequalities for the norm of a matrix,[3]

$$N(M^*) \leq 2N(\lambda)\{1 + N[(I + L + R)^{-1}]N(I + L + R)\}.$$

Hence if an estimate of $N\{(I + L + R)^{-1}\}$ may readily be obtained, the theorem gives a practical test for the convergence to 0 of the Seidel iterants of a nearly symmetric matrix.

We proceed with the proof of the theorem. Let $x^{(0)}$ be an arbitrary vector; let

$$x = B^n x^{(0)}, \quad y = Bx, \quad z = x - y.$$

We suppose $I - B$ to be non-singular, and so obtain

$$x = (I - B)^{-1}z, \quad x + y = (I + B)(I - B)^{-1}z = (I - B)^{-1}(I + B)z.$$

Since C is symmetric,

$$(1) \qquad x'Cx - y'Cy = (x' - y')C(x + y) = z'C(I - B)^{-1}(I + B)z.$$

Since $B = -(I + L)^{-1}R$, and $C = A + \lambda' - \lambda$, substituting in (1) we get

$$(2) \quad x'Cx - y'Cy = z'(I + L + R - \lambda + \lambda')(I + L + R)^{-1}(I + L - R)z.$$

We interrupt the proof to state a lemma:

Lemma. *If C is a symmetric matrix, $y = Bx$, $x'Cx - y'Cy > 0$, then* $\lim_{n\to\infty} B^n = 0$ *or* $\neq 0$, *according as C is positive definite or not.*

A formal proof of the lemma is given elsewhere.[4]

If now $\lambda = 0$ then (2) gives

$$x'Cx - y'Cy = z'z > 0 \text{ if } z \neq 0.$$

Since $z = (I - B)x$ and $I - B$ is not singular, $z \neq 0$ if $x \neq 0$. By the lemma the known result for the convergence of the Seidel iterants then follows immediately.

If A is not symmetric, we may continue proving the theorem as follows. For any matrix P, $z'Pz = \frac{1}{2}z'(P + P')z$, and $\frac{1}{2}(P + P')$ is a symmetric matrix. Applying this result to the matrix on the right of (2) we get $x'Cx - y'Cy = z'Mz$, where M is the matrix defined in the enunciation of the theorem. The theorem now follows from the lemma.

If $A = D + L + R$, and $D \neq I$, all $a_{ii} > 0$, the theorem is still true if we generalize somewhat the definitions of C and M. Let δ be the diagonal matrix whose elements are the positive square roots of the elements of D. Then $\delta^2 = D$. The Seidel matrix of A is then

$$B = -(D + L)^{-1}R = -(\delta^2 + L)^{-1}R = -\delta^{-1}(I + \delta^{-1}L\delta^{-1})^{-1}\delta^{-1}R.$$

If we write $B^* = \delta B\delta^{-1}$, the characteristic roots of B^* are those of B. Hence the B iterants will converge to zero or not according as the B^* iterants converge to zero or not. Again $B^* = -(I + \delta^{-1}L\delta^{-1})^{-1}\delta^{-1}R\delta^{-1}$. Hence B^* is the Seidel iteration matrix for $I + \delta^{-1}L\delta^{-1} + \delta^{-1}R\delta^{-1}$.

Now, if $I + L + R$ is symmetric, so is $I + \delta^{-1}L\delta^{-1} + \delta^{-1}R\delta^{-1}$. The theorem may thus be extended to the more general case provided that in the definitions of C, B, λ and M, we replace L and R by $\delta^{-1}L\delta^{-1}$ and $\delta^{-1}R\delta^{-1}$ respectively.

If $A = D + L + R$ is a symmetric matrix, the condition that A is positive definite in order that the Seidel iterants of its associated matrix converge to zero is not only sufficient but, in the case when the elements of D are all positive, is also necessary.[5] For we observe that in the case $D = I$ this result is a special case of the theorem. If $D \neq I$, we form the matrix $A^* = \delta^{-1}A\delta^{-1}$ where δ is defined as above. If now M is any leading minor of A and M^* the corresponding minor of A^*, and δ_m the elements in δ corresponding to M, then $M^* = \delta_m^{-1}M\delta_m^{-1}$ and so $\det(M^*) = (\det \delta_m^{-1})^2 \det(M)$.

If the elements in D are all positive, $(\det \delta_m^{-1})^2$ is positive and $\det(M^*)$ and $\det(M)$ are both of the same sign. Hence A and A^* are both positive definite, or neither is positive definite. Since the Seidel iterants of the matrices corresponding to A and A^* either both converge to zero or neither does, and the leading diagonal in A^* is I, the result for A follows from what has been proved for A^*.

I wish to thank OLGA TAUSSKY-TODD for the encouragement I received from her in the course of the preparation of this paper. I also wish to thank my colleague, S. E. CRUISE, for reading and checking the manuscript.

P. STEIN

University of Natal
Durban, Union of South Africa

[1] See, for example, HILDA GEIRINGER, "On the solution of systems of linear equations by certain iteration methods," Reissner Anniversary Volume, *Contributions to Applied Mechanics*, Ann Arbor, 1948, p. 365–393.

[2] This is Seidel's result. For a recent proof see R. VON MISES & HILDA GEIRINGER, "Praktische Verfahren der Gleichungsauflösung," *Zeitschr. f. angew. Math. u. Mech.*, v. 9, 1929, p. 58–77, 152–164.

[3] See, for example, p. 1044 in JOHN VON NEUMANN & H. H. GOLDSTINE, "Numerical inverting of matrices of high order," Amer. Math. Soc., *Bull.*, v. 53, 1947, p. 1021–1099.

[4] P. STEIN, "Some general theorems on iterates," to appear in NBS *Jn. of Res.*

[5] A different proof of this result is given by E. REICH, "On the convergence of the classical methods of solving linear simultaneous equations," *Ann. Math. Stat.*, v. 20, 1949, p. 448–451.

BIBLIOGRAPHY Z–XVII

1. ANON., "Conversion between analogue and digital data," ONR London, *Technical Report* ONRL-73-50, July 1950, 4 pages.

This brief report from the Scientific Liaison Officer of the American Embassy in London, England, summarizes a symposium held at the Military College of Science, Shrivenham, 5–6 July, 1950. The following prepared papers were presented and discussed:

(1) "A method of telerotation by pulse code modulation," by R. H. BARKER of the Signals Research and Development Establishment, Christchurch, Hants.,

(2) "The production and use of binary coded discs for data transmission," by W. S. ELLIOTT, R. C. ROBBINS, and D. S. EVANS of the Research Laboratories of Elliott Bros., Borehamwood,

(3) "Telerotation by parallel binary codes," by E. A. JOHNSON of the Radar Research and Development Establishment, Great Malvern, Worcs.,

(4) "A servo system operating on discontinuous information," by C. HOLT-SMITH and D. LAWDEN of the Faculty of Instrument Technology, Military College of Science, Shrivenham,

(5) "Note on digital computing for the extrapolation of discontinuous information," by G. C. TOOTILL of the Faculty of Instrument Technology, Military College of Science, Shrivenham,

(6) "Some methods of conversion of data from analogue (shaft rotation) to digital form," by K. V. DIPROSE of the Royal Aircraft Establishment, Farnborough, Hants.,

(7) "The electromagnetic cam: a technique for wave manipulation using the nonlinear properties of magnetic materials," by W. LAWRENCE of the Signals Research and Development Establishment, Christchurch, Hants.

Copies of these papers have been forwarded to Code 250, Office of Naval Research, Washington 25, D. C., and the proceedings of the Symposium (except for these papers) will be prepared and distributed by the Ministry of Supply.

The different systems described for conversion between shaft rotation and digital codes had different chief features such as compactness, accuracy of one part in 20,000, or a three microsecond digit time. The prediction problem was treated both by direct use of the discrete data and by polynomial approximation to the prediction function.

R. D. ELBOURN

NBSCML

2. ANON., "Desk-sized electronic computer developed by Northrop," *Machinery*, v. 57, Mar. 1951, p. 168.
Short expository article on the MADDIDA.

3. ANON., *Digital Computer Newsletter*, ONR Mathematical Sciences Division, v. 3, April 1951, 6 pages.
The contents are as follows:

1. The ERA 1101 Computer
2. Moore School Automatic Computer
3. The Institute for Advanced Study Computer
4. Aberdeen Proving Ground Computers: The ENIAC, The EDVAC, The ORDVAC
5. National Bureau of Standards Western Automatic Computer (SWAC)
6. National Bureau of Standards Eastern Automatic Computer (SEAC)
7. Project Whirlwind
8. Relay Digital Computer, Imperial College, Univ. of London
9. Zuse Computer Model IV, at Zurich, Switzerland
10. Data Handling and Conversion Equipment: Stavid Engineering Data Conversion Equipment, Signal Corps Angular Position Encoders, and Zatacoding

4. ANON., *Digital Computer Newsletter*, ONR Mathematical Sciences Division, v. 3, July 1951, 6 pages.
The contents are as follows:

General Purpose Computers

1. Burroughs Laboratory Computer
2. UNIVAC Dedication
3. The Raytheon Digital Computer
4. The National Bureau of Standards Eastern Automatic Computer
5. Whirlwind I (M.I.T.)
6. The ORDVAC (Univ. of Ill.)
7. Moore School Automatic Computer (MSAC)
8. The Institute for Advanced Study Computer

Special Purpose Computers

1. The CRC 101 Digital Differential Analyzer
2. The MADDIDA

Auxiliary and Conversion Equipment

1. Flying Typewriter
2. Conversion and Display Equipment

5. ANON., "Guided-missile computer," *Electronics*, v. 24, Apr. 1951, p. 138.
This is an expository paper which describes the hybrid analog-digital computer being developed at Project Typhoon, Special Devices Center of ONR, Port Washington, N. Y.

6. T. J. CONNOLLY, S. P. FRANKEL, & B. H. SAGE, "Application of automatic digital computing methods to the production of phase behavior," *Electrical Engineering*, v. 70, Jan. 1951, p. 47.

Punched card methods using the type 604 calculating punch were developed to predict the thermodynamic properties of homogeneous and heterogeneous multicomponent hydrocarbon systems by solving the Benedict equation of state. Methods described permit evaluating the composition and the molal values of volume, entropy, and enthalpy of the coexisting phases at one heterogeneous state in about one hour of computing, or for a single homogeneous phase in about ten minutes. Thus computing particular values as required is feasible whereas tabulating completely the thermodynamic properties of multicomponent heterogeneous systems is not. A bibliography of 36 titles is included.

R. D. ELBOURN

NBSCML

7. D. H. GRIDLEY & B. L. SARAHAN, "Design of the Naval Research Laboratory Computer," *Electrical Engineering*, v. 70, Feb. 1951, p. 111.

It is expected that the Naval Research Laboratory Electronic Digital Computer (NAREC) will be working by January 1952 and will be in productive operation within six months following. It will have 1024 words of electrostatic storage and 2048 of magnetic drum storage. Words will consist of 44 binary digits and sign; instruction words will be interpreted as pairs of one-address orders. Input and output will be on magnetic tape. The machine will have a multiplication time of 300 microseconds.

MICHAEL MONTALBANO

NBSCL

8. LAWRENCE P. LESSING, "The electronics era," *Fortune*, v. 44, July 1951, p. 79–83, 132–138.

Briefly mentioned in this article are the important recent developments in the computer field. To quote from the paper: "It may be that the most significant accomplishment for electronics for the future is its ability to count."

9. D. M. MCCALLUM & J. B. SMITH, "Mechanized reasoning, logical computers and their design," *Electronic Engineering*, v. 23, Apr. 1951, p. 126–133.

A small electrical computer, built at Ferranti, Ltd. in Edinburgh, to evaluate Boolean propositional functions of up to seven binary-valued variables is described. A function is set up by plugging together logical connective boxes which contain relays, selenium diodes, and indicator lamps. The computer then tries automatically in sequence the 128 combinations of values of the variables and stops on any combination for which the function is "true."

One possible extension described is particularly interesting because a computer using it might find reasonably quickly a solution for a function of so many variables that systematic scanning would take a prohibitively long time; moreover, the process appears to be more like that used by the

human brain. The method is to assume some set of values of the variables, which usually will not be a solution, then to identify those variables whose values appear to be responsible for the false result and, by feedback, to cause them to change more or less at random until a true result is achieved. A scheme for avoiding circular sequences of changes is suggested.

NBSCML R. D. ELBOURN

10. AMBROS P. SPEISER, *Entwurf eines Elektronischen Rechengerates*, Inst. f. angew. Math., *Mitt.*, Zürich, No. 1, 1950.

This is a description of the logical design of a digital computer intended to be built at low cost and to operate at relatively low speed, but otherwise to equal the performance of most of the modern automatic machines. No new engineering development is contemplated; use is made only of existing and well-tested components, especially of those developed by H. H. Aiken. The memory organ is a magnetic drum holding 1200 words, each consisting of sign and 12 decimal digits. The average access time is 16 milliseconds. The arithmetic organ operates serially, with a pulse repetition rate of 40 k.c. The addition time is .8 ms; the average multiplication time 30 ms. Multiplication is accomplished by repeated addition, division by an original modification of the repeated subtraction process. Instructions are written in a single-address system, pairs of addresses being stored in the memory as single words. The arithmetic unit and part of the control are electronic, but address selection is performed by relays. The design calls for 1000 vacuum tubes and 500 relays in the entire machine. Teletype equipment is used for input and output, without provision for external storage. There is no automatic checking, except that the machine stops when certain "forbidden" pulse combinations occur.

NBSCL FRANZ L. ALT

11. H. W. SPENCE, "Systematization of tube surveillance in large scale computers," *Electrical Engineering*, v. 70, July 1951, p. 605–608.

Ninety per cent of the service interruptions of ENIAC at Aberdeen Proving Ground are caused by vacuum-tube failures. This paper describes the first five months' operation of a new tube-surveillance program. The average life of some 5000 tubes discarded varied among ten types from 5,500 to 12,500 hours. Much of the difference between types is attributed to different operating and testing conditions. For most types low emission is the predominant cause of failure, especially so in normally OFF tubes. If only low-emission failures are considered, the life of tubes used in the ON condition is better than that of the same type used in the OFF condition by 50 per cent or more. Poor cutoff, short circuited elements, and burned out filaments may be considered accidental failures because life terminated by these averaged less than half as long as that for low-emission failure.

Because it contains the first extensive tube life data appropriate to computer design, this report is especially welcome and further results are eagerly awaited.

NBSCML R. D. ELBOURN

NEWS

Department of the Air Force and National Bureau of Standards.—A symposium on Linear Inequalities and Programming was held in Washington, D. C., on June 14, 15, and 16, 1951. This meeting was sponsored by the Department of the Air Force and the National Bureau of Standards. The program was as follows:

Thursday, June 14, Morning	
Linear inequalities	J. TODD, *Chairman*, USAF
Welcoming talk	Gen. F. J. DAU, USAF
Theorems of alternatives for pairs of matrices	A. W. TUCKER, Princeton Univ.
Remarks on the history of work on linear inequalities	T. S. MOTZKIN, NBSINA
Optimization in unitary spaces	E. W. BARANKIN, Univ. of Calif. and RAND Corp.
Polyhedral cones	A. CHARNES, Carnegie Inst. of Tech.
Geometrical significance of the simplex method	J. P. MAYBERRY, Princeton Univ.
Discussion	
Thursday, June 14, Afternoon	
Applications	T. C. KOOPMANS, *Chairman*, Cowles Commission, Univ. of Chicago
Choice models and non-choice models	W. W. LEONTIEF, Harvard Univ.
Research program of Project SCOOP	M. K. WOOD, USAF
Discussion	
Friday, June 15, Morning	
Computational theory and techniques	G. B. DANTZIG, *Chairman*, NBSCL
Welcoming talk	Gen. F. J. DAU, USAF
New techniques for linear inequalities and optimization	T. S. MOTZKIN, NBSINA
Methods of solving linear equations	G. FORSYTHE, NBSINA
Gradient methods in Lagrangian problems and their game theoretical interpretation	L. HURWICZ, Univ. of Illinois, and Cowles Commission, Univ. of Chicago
Properties of Leontief matrices	Y. K. WONG, Princeton Univ.
Problems of formulation	P. A. SAMUELSON, *Chairman*, MIT and RAND Corp.
Efficiency prices for decentralized decisions	T. C. KOOPMANS and G. DEBREU, Cowles Commission, Univ. of Chicago
Suitability scales for allocation problems	J. L. HOLLEY, USAF
Personnel classification	D. F. VOTAW, Jr., Yale Univ., and A. ORDEN, USAF
Design of an optimal battery of tests	M. A. WOODBURY, Princeton Univ.
Saturday, June 16, Morning	
Nonlinear and linear problems	E. D. SCHELL, *Chairman*, USAF
Theory of the transportation problem	M. M. FLOOD, RAND Corp.
Convex programming	D. W. BLACKETT, Princeton Univ.
Least ballast shipping required to meet a specified shipping program	I. HELLER, Logistics Research, George Washington Univ.
Problem of contract awards	L. GOLDSTEIN, USAF
A gasoline blending problem	W. COOPER, Carnegie Inst. of Tech.
Discussion	

Saturday, June 16, Afternoon

Computational theory and techniques	F. L. ALT, *Chairman*, NBSCL
Matrix inversion	H. H. GOLDSTINE, Inst. for Advanced Study, Princeton Univ.
A unified technique for matrix inversion, linear inequalities, games, optimization	A. ORDEN, USAF
A short proof of the dual theorem	G. B. DANTZIG, USAF
The projective method and computational experience	C. B. TOMPKINS, Logistics Research, George Washington Univ.
Discussion	

Eckert-Mauchly Division, Remington Rand Inc.—The first UNIVAC, which was constructed for the Bureau of the Census, under NBS contract with Remington Rand, was delivered by EDWARD U. CONDON, Director of the NBS, to ROY V. PEEL, Director of the Bureau of the Census, on June 14th.

During the elaborate ceremonies which took place in Philadelphia on that occasion, the UNIVAC was busy on a typical task of the Bureau of the Census—the tabulation of information about JOHN Q. CITIZEN. In one-sixth of a second, the machine classified an individual according to place of residence, sex, color, age group, nationality, citizenship, extent of education, educational activity, farming activity, government activity, marital status, domestic status, veteran status, employment status, occupational group, industrial group, earning capacity, etc. For each county, the resulting tabulations were printed out by the UNITYPER under the appropriate headings. Any interested guest at the dedication ceremonies had the opportunity to check the crossfootings testing the accuracy of the machine's operations.

Even at the amazingly rapid rate of six persons per second, the classification of 150,000,000 individuals and the preparation of the various and sundry tabulations is too formidable a task for a single machine. It is planned therefore to continue the use of punched card equipment as heretofore, in conjunction with the tabulation by the UNIVAC of the residents of a few selected states. The Bureau of the Census estimates that about ten per cent of the total task will be accomplished by electronic equipment by the time the 1950 tabulations will have been completed.

During the morning of June 14th, a press seminar was held at which the following program was presented:

Welcome	ALBERT GREENFIELD, President, Chamber of Commerce, Philadelphia
Description of the UNIVAC	J. PRESPER ECKERT, Eckert-Mauchly Division, Remington Rand, Inc.
Description of Demonstration Problem	JAMES L. MCPHERSON, Bureau of the Census
Demonstration of the UNIVAC	S. N. ALEXANDER, NBS
Press Forum	

During the afternoon the dedication ceremonies were held. The program was as follows:

Chairman	ALBERT GREENFIELD, President, Chamber of Commerce, Philadelphia
Economic Implications	JAMES H. RAND, President, Remington Rand, Inc.
Official Delivery of the UNIVAC	EDWARD U. CONDON, Director, NBS, and ROY V. PEEL, Director, Bureau of the Census
Dedication	CHARLES SAWYER, Secretary of Commerce