R E P O R T   R E S U M E S

ED 019 889 EM 006 693

REVIEWS OF FILMS, REPORT OF SOME REVIEWING COMMITTEES.
NATIONAL COUNCIL OF TEACHERS OF MATHEMATICS INC.
EDRS PRICE MF-$0.25 HC NOT AVAILABLE FROM EDRS. 29P.

DESCRIPTORS- *MATHEMATICS, *COLLEGE MATHEMATICS, *SECONDARY SCHOOL MATHEMATICS, *MATHEMATICS INSTRUCTION, *INSTRUCTIONAL FILMS, RESOURCE GUIDES, REFERENCE MATERIALS, TEACHER TRAINING EDUCATION, NATIONAL COUNCIL OF TEACHERS OF EDUCATION

A TITLE AND SERIES TITLE LIST OF SECONDARY AND TEACHER EDUCATION MATHEMATICS FILMS GIVES A REVIEW OF EACH FILM, WITH TITLE, LENGTH, PRICE, SOUND/SILENT, BLACK AND WHITE/COLOR, RECOMMENDED LEVEL (ELEMENTARY, JUNIOR HIGH SCHOOL, SENIOR HIGH SCHOOL, JUNIOR COLLEGE, SENIOR COLLEGE, OR ADULT), AND AVAILABILITY OF TEACHERS GUIDE OR STUDENT MANUAL. EACH FILM WAS REVIEWED BY ONE OF THREE COMMITTEES OF FIVE OR SIX HIGH SCHOOL AND COLLEGE MATHEMATICS TEACHERS, WITH EMPHASIS ON CONTENT, EFFECTIVENESS, QUALITY, AND POSSIBLE USES. A SUBJECT INDEX AND A LIST OF FILM DISTRIBUTERS WITH ADDRESSES ARE INCLUDED. THIS DOCUMENT IS A REPRINT FROM "THE MATHEMATICS TEACHER," DECEMBER 1963 AND IS AVAILABLE FOR $0.40 FROM THE NATIONAL COUNCIL OF TEACHERS OF MATHEMATICS, 1201 SIXTEENTH STREET, N.W., WASHINGTON, D.C. 20036. (BB)

A-II

REVIEWS OF FILMS

EM 06693

REPORT OF SOME REVIEWING COMMITTEES

Reprinted from
The Mathematics Teacher
December 1963

NATIONAL COUNCIL OF TEACHERS OF MATHEMATICS
1201 Sixteenth Street, N.W., Washington, D.C. 20036

REVIEWS OF FILMS

REPORT OF SOME REVIEWING COMMITTEES

Reprinted from
The Mathematics Teacher
December 1963

NATIONAL COUNCIL OF TEACHERS OF MATHEMATICS
1201 Sixteenth Street, N.W., Washington, D.C. 20036

Reviews of films

A REPORT OF SOME REVIEWING COMMITTEES

Some films prepared for use in secondary school mathematics classes and in teacher education classes are listed and reviewed.

DURING 1962, three committees of high school and college mathematics teachers met on several occasions to review many of the available mathematics films for junior and senior high school. These meetings were financed by the National Council of Teachers of Mathematics. The results of these reviewing sessions are to be found on the following pages.

Originally, the project director, Joseph A. Raab, contacted approximately one hundred film producers and suppliers, and more than two hundred and fifty films were supplied for reviewing. There are, of course, many more mathematics films available from various sources, but those films which were viewed by the committees seem to constitute a good sample.

Each of the three committees was made up of five or six mathematics teachers who were instructed to examine the films with regard to their mathematical content, pedagogical effectiveness, and technical quality. In addition, where possible, particular uses for each film were to be indicated as well as the general level of the film. In general, the committees followed through with these instructions although it was deemed necessary to edit the reviews somewhat in order to achieve some uniformity in style and to condense some reviews.

Thus, each review represents the composite opinion of five or six mathematics teachers. Generally, they have placed more importance on the mathematical content of the film than on its technical aspects. Such emphasis is probably well justified since a film which is mathematically correct will be useful in the classroom in spite of some technical faults while no amount of technical skill can make a useful film from one that is mathematically incorrect. Several committee members have commented that the mathematical quality of many of the films is poor even in cases where a mathematics consultant is listed in the production. This is an unfortunate situation which might be remedied by more effective use of these consultants in the actual production of the film. Moreover, there seem to be relatively few films designed to illustrate a particular topic in less than ten minutes, say, and a great many films thirty minutes long, each covering a great many topics. Animation seems to be used less extensively and effectively than one might expect in view of the number of topics in mathematics which involve motion of some sort. Many members felt that short films on the history of mathematics would be of particular value to the mathematics teacher. In short, the consensus of the reviewers seems to be that more mathematics films should be designed to do tasks that the ordinary classroom teacher cannot do effectively, and fewer films designed to "teach" an entire course.

The following mathematics teachers constituted the three committees. Their concerted efforts, constant attention to the job, and open minded attitude are greatly appreciated.

Subject index of films

List of film distributors

Association Films, Inc.
347 Madison Avenue
New York 17, New York

Cenco Fducational Films
1700 Irving Park Road
Chicago 13, Illinois

Colburn Film Distributors
1122 Central Avenue
Wilmette, Illinois

Coronet Instructional Films
65 East South Water Street
Chicago 1, Illinois

Educational Research Council of Greater Cleveland
Rockefeller Building
Cleveland 13, Ohio

Encyclopaedia Britannica Films, Inc.
1150 Wilmette Avenue
Wilmette, Illinois

Indiana University
Department of Astronomy
Bloomington, Indiana

International Film Bureau
332 South Michigan Avenue
Chicago 4, Illinois

McGraw-Hill
330 West 42nd Street
New York 36, New York

Modern Learning Aids
3 East 54th Street
New York 22, New York

State University of Iowa
Bureau of Audio-Visual Instruction
Extension Division
Iowa City, Iowa

University of Michigan Films
4028 Administration Building
Ann Arbor, Michigan

Walt Disney Productions
Educational Films Division
Burbank, California

Alphabetical listing and reviews of films

Addition and Subtraction of Rational Numbers. See Intermediate Algebra Series.

Addition Formulas and DeMoivre's Theorem. See Trigonometry Series.

* The following is a key to the symbols used:

sd: sound	jh: junior high school
si: silent	sh: senior high school
bw: black and white	jc: junior college
co: color	sc: senior college
el: elementary	a: adult

guide: teacher's guide available
manual: student manual available

Advanced Algebra Series. 20 films, 1960. sd, bw; sh, jc; tchrs. of sh.*
Modern Learning Aids, $3000.†

Historical Introduction to Algebra. 27 min., $150; jh, sh, jc; tchrs. of jh, sh.

This delightful first film in the advanced algebra series is ably narrated and would be of interest to algebra students at almost any level.

Algebra is discussed as a generalization of arithmetic and is shown to have gone through the rhetorical, syncopated, and symbolic stages of development. Some great books in the story of mathematics are discussed with photos of the Rhind Papyrus, copies of the Diophantos *Arithmetic*, and *Al-Jabr* (from which "algebra" is derived) being shown.

An excellent discussion of a cultivation problem brings out questions of manipulation which are to be answered in later films.

Five Fundamental Postulates of Algebra. 30 min., $150; sh, jc; tchrs. of sh.

Five fundamental postulates are given, namely: commutative and associative laws for addition and multiplication, and the distributive postulate for multiplication over addition. It is emphasized that these postulates are not theorems. An interpretation of the process of division and the implications of dividing by zero are discussed, along with the ideas of least common multiples and least common denominators.

More grouping symbols need to be used in connection with the fundamental postulates. The principles for multiplying by one and for adding zero are called obvious. The discussion of division by zero is not carefully done. The geometric model used to show that the product of two negative numbers is a positive number is very confusing. The film will be fair for review but is poor for teaching these ideas.

Introduction to Factoring. 30 min., $150; sh, jc; tchrs. of sh.

After a review of the five fundamental postulates for algebra, factoring forms are specifically stated for monomials with common terms, difference of two squares, and quadratic perfect squares. The ideas of factoring algebraic expressions are developed with reference to the factoring forms. A discussion of errors that occur in cancelling is related to factoring.

Although the review of the previous film on postulates is good, the lecturer fails to use these postulates in the discussion on factoring. These factoring techniques involve memorization of certain forms. Stress is placed on the idea that factorization is simply the mastery of certain rules which are to be applied in nine out of ten cases! The third form of factoring is given as $A^2 \pm 2AB + B^2 = (A+B)^2$ which is incorrect.

† The prices listed are either purchase price or long-term lease. Rental prices may vary depending on source and for this reason are not listed.

Standard Techniques of Factoring. 30 min., $150; sh, jc; tchrs. of sh.

The following standard forms to be used for factoring are stated: quadratic perfect squares, quadratic nonperfect squares, sum of cubes, and difference of cubes. The lecturer discusses "splitting the middle term" as a method of handling forms involving the quadratic nonperfect square. Certain wrong techniques which are sometimes used in cancelling are identified.

Apparently, these factoring forms are to be memorized. The formula for factoring a general quadratic form is grossly oversimplified. The display of "wrong techniques of cancelling" is not good.

Simplifying Complex Fractions. 30 min., $150; sh, jc; tchrs. of sh.

Mechanical operations for handling complex fractions are discussed including: (1) in simple cases, invert the denominator and multiply this times the numerator; (2) simplify the numerator then simplify the denominator before using rule (1). Cancelling is mentioned as a factoring technique when both numerator and denominator have been expressed in simplest terms. Certain examples are provided to show the errors that arise when cancelling across addition or subtraction is performed.

The simplifying of complex fractions is generally void of any logical or mathematical reasoning. That "invert and multiply" is equivalent to the "logical" procedure is shown by giving one example. The language used in this entire film is extremely sloppy.

Linear Equations in one Unknown. 30 min., $150; sh, jc; tchrs. of sh.

The solution of linear equations is introduced by considering the "distance-rate-time" problem in great detail. A small portion of the film is devoted to the transformation of equations to equivalent equations. Extraneous roots of linear equations are touched upon.

Good techniques are used for setting up an equation although the point of view toward equation solving is old-fashioned. Some checking of results is shown but often the result is simply shown to be reasonable. The treatment of extraneous roots is not clear and too much time is spent on mechanical details which really do not help a student understand how to solve linear equations.

Introduction to Simultaneous Equations. 30 min., $150; sh, jc; tchrs. of sh.

The solution of two simultaneous linear equations is introduced by a verbal problem. Two methods of solution are discussed—comparison and determinants.

The treatment is definitely not introductory in nature as the title suggests. The "method of comparison" is good but the use of determinants is hastily done. The emphasis on mechanical operations is distinctly inappropriate and too much mathematics is passed by as being obvious.

Determinants and Cramer's Rule. 30 min., $150; sh, jc; tchrs. of sh.

Two simultaneous linear equations are solved using determinants of order two followed by an analysis of three simultaneous linear equations from which Cramer's rule is derived. Determinants of order three are expanded by a method of "bordering" using first and second columns.

The algebraic ideas are grossly oversimplified and there is too much emphasis on mechanical techniques using determinants. The student may be left with the impression that the use of determinants is always easy and the best technique for solving such systems.

Determinants of Any Order. 30 min., $150; sh, jc; tchrs. of sh.

A general theorem for expanding a determinant by minors of rows or columns is stated and the notion of minors is fully discussed. Careful attention is given to the general applicability of Cramer's rule to determinants of any order.

Although too much stress is placed on techniques and not enough on understanding, the explanation of minors is very good. Such phrases as "throw the zeros in the first column" indicate the nature of the terminology used.

Introduction to Quadratic Equations. 30 min., $150; sh, jc; tchrs. of sh.

Verbal problems are proposed whose solutions involve quadratic equations. The problems are actually worked out and the emphasis is on the solution of the quadratic equation by completing the square. A derivation of the quadratic formula follows.

The word problems used to give rise to quadratic equations are good. The method of completing the square is reasonably well done but the emphasis is on the quadratic formula which is to be memorized.

Solving Problems with the Quadratic Formula. 30 min., $150; sh, jc; tchrs. of sh.

The quadratic formula is used to solve a quadratic equation whose roots are irrational. The square root of a number is approximated by the square root algorithm, by the slide rule, and from tables of squares and square roots. Emphasis is placed on location of the decimal point. The use of tables in approximating cube roots concludes the film.

The algorithm for approximating square roots is given purely as a mechanical process without appropriate foundation. Emphasis is on memorization of the quadratic formula with no mention of the principles underlying it. The simplifying of radicals is done well. The film title is misleading since the major portion of the work is devoted to handling square roots.

Complex Numbers and Roots of Equations. 30 min., $150; sh, jc; tchrs. of sh.

Analysis of the discriminant includes the cases where the discriminant is greater than zero, equal to zero, and less than zero. Definitions of the square root of negative one, a complex number, and an imaginary number are given. The representation of complex numbers in the number plane is described. The film concludes with a discussion of the nature of the roots of systems of quadratic equations.

Although i is defined, very little is done with operations with complex numbers. The complex number plane is handled in a very sketchy manner. The discussion of the nature of the discriminant of a quadratic equation is satisfactorily handled.

Introduction to Graphs of Equations. 30 min., $150; sh, jc; tchrs. of sh.

Systems of quadratic equations are discussed along with a system of a quadratic and a linear equation. The use of graphs in interpreting systems of quadratic functions is discussed.

The technique of substituting a linear equation in a quadratic is poorly done. No satisfactory definition of a "quadratic system" is given.

Graphs of Quadratic Equations. 30 min., $150; sh, jc; tchrs. of sh.

Attention is given to the graphical interpretation of circles, ellipses and parabolas. The discussion is an extension of the presentation in the preceding film. The technique of completing the square is employed in graphing these functions in order to determine a set of transformed axes to reduce the functions to a standard form.

Entirely too much is attempted in this film with the result that the viewer may not understand the basic differences between the graphs of circles, ellipses, and parabolas. The use of the technique of completing the square is satisfactory but is overwhelmed in the bulk of material.

Theory of Equations and Synthetic Division. 30 min., $150; sh, jc; tchrs. of sh.

Several parabolic functions are graphed using the technique of completing the square. Finding the roots of a cubic equation motivates the introduction of synthetic division which is subsequently used as a test for roots.

Not enough is made of the role of the Factor Theorem and Remainder Theorem. The role of the "missing term" and the linearity of the divisor in synthetic division is not made clear. The viewer is likely to be confused by the film.

Solution of Equations Beyond the Second Degree. 32 min., $150; sh, jc; tchrs. of sh.

A quartic equation is used to motivate discussion of the Rule of Signs as well as other rules which enable a student to carry out a limited search for rational roots.

Unfortunately, the discussion here is almost completely centered on rules with very little analysis of the relationship between coefficients and roots of the polynomial equations. The result is a film on manipulation and technique.

Permutations and Combinations. 30 min., $150; sh, jc; tchrs. of sh.

The meaning of permutations and combinations is discussed. The basic formulas for number of permutations of n different objects taken r at a time, without repetitions; permutations of a set of objects not all different; and number of combinations of n different objects taken r at a time are derived and discussed. These formulas are then applied to problems with the admonition that restrictions are often necessary in order to carry through with these formulas.

For a person who is already well acquainted with these topics the film should be acceptable. The derivations of the basic results are not particularly well done. We suggest the film be used for reviewing only.

Nature of Logarithms. 30 min., $150; sh, jc; tchrs. of sh.

A complicated multiplication problem is used as motivation for the study of logarithms. Mantissa and characteristics are discussed carefully. The emphasis is on the base ten. The use of positive mantissas with negative characteristics is discussed.

Although the development is consistent with the traditional approach, we feel the stress should be placed more heavily on the underlying mathematical notions.

Using Logarithms in Problems. 30 min., $150; sh, jc; tchrs. of sh.

Logarithms with arbitrary bases are discussed including the definition of $\log_a c = b$ as $a^b = c$. Such logarithms are used to solve problems.

The unfortunate inference that the viewer will draw is that the primary purpose of logarithms is to simplify computation. The film will be satisfactory for review purposes.

Computing Logarithms from Arithmetic and Geometric Series. 30 min., $150; sh, jc; tchrs. of sh.

A short history of logarithms is given as well as an explanation of the derivation of the tables. Interpolation is discussed at some length as well as other topics in the use of tables of mantissas.

The historical introduction for the evaluation of logs for tables is good. The tie-in with the previous film, *Nature of Logarithms,* is timely. The title of the film is misleading since very little discussion centers on these series. The major use of the film should be for review.

ADVENTURES IN NUMBER AND SPACE SERIES. 9 films, 1958. sd, bw; el, jh, sh, a; tchrs. of el, jh; Association Films, Inc., $1,250.

The series consists of nine films originally produced for television. Although a variety of topics are presented, the series is made contiguous by the presence of Mr. Bill Baird and his marionettes. The resulting vehicle is very effective as a device to arouse interest in mathematics. A note of levity is constantly provided by the discourse, antics, and predicaments of the marionettes. Mr. Baird provides lectures and, at times, problem solving techniques for getting the marionettes out of their scrapes.

The props and devices used in the film will not generally be found in the classroom. Thus, the individual films provide much in the way of motivation and introduction to new topics that would not ordinarily be possible. On the other hand, very little teaching of substantial subject matter will be accomplished directly from use of these films due to the elementary level of treatment. However, the amount of interest generated should easily carry over to regular classroom instruction and give real support to more advanced levels of treatment.

The series has been very well edited on the whole. We found little which could be added or deleted to improve the resultant product. Camera technique is excellent throughout the series and the result is a series which is technically very good.

The most receptive audiences for the series would probably be found at the seventh- through ninth-grade levels. At these levels the films will be good motivational devices for the introduction to algebra, geometry, trigonometry, statistics, and other areas. However, we suggest caution in relying on them as sources of mathematics content as indicated above. The mathematical notions treated are elementary and are not always well developed or connected.

How Man Learned to Count. 30 min., $150; el, jh, a.

The following notions are dealt with: the elementary reasons and ideas for counting, the Egyptian numeration system up to thousands, the Roman and Babylonian numeration systems, the counting board, the abacus and modern calculators, and the origin of zero.

The reason for counting and having a numeration system is motivated through a classical problem involving cavemen bartering for an ax in terms of "one-two-and a heap" of shells. The difficulties in some notational systems are exemplified by a Roman general attempting to reach a decision about the total number of spears he needs to outfit his army when he knows the number of columns of men and the number of men in each column.

The major portion of the film is devoted to developing the rules governing the counting board. Examples are given and solved by addition using marbles (stones) and grooves in a board (sand). It is mentioned that the counting board was the beginning of the modern decimal system. The abacus is explained and problems are solved on it in competition with a modern electric calculator.

An overwhelming majority of historians believe that the Arabs had little if anything to do with the invention of a zero symbol. Their chief contribution was transmitting the zero symbol to the Western World.

Quicker Than You Think. 30 min., $150; el, jh, a.

This film is essentially concerned with a development of the binary system, done rather quickly. The motivation for the study of such a system is provided by a chart from which a person's age might be deduced when the person selects the various columns in which his age appears. The presentation is at first a bit awkward, but as the film progresses, there is a good set of devices used to introduce the notion of the binary system. Modern computing machines are demonstrated by visiting a laboratory in which a Westinghouse computer is used, and a brief description of the mechanics of its operation is given.

Mysterious X. 30 min., $150; el, jh, a.

This film gives good motivation for algebra. Examples of the use of algebra in life are given with the use of formulas exemplified by: The "cricket chirp" formula for temperature, Newton's laws, and Einstein's $E=mc^2$.

The presentation defines algebra as a set of laws governing numbers and states some of the laws—for example, the commutative laws. Solution of equations is poorly done since it is being done intuitively rather than by taking advantage of the laws previously considered and named. The word "variable" is not well defined.

This film might be usable at the fifth- through ninth-grade level.

What's the Angle. 30 min., $150; el, jh, a.

What's the Angle is an historical introduction to the notions of a right triangle and some of the consequences thereof. The film is motivated by the marionettes' confusion over the shape of a baseball diamond, that is, its being a square rather than what they thought a diamond should look like. This leads into the notion of a right triangle and from this Mr. Baird proceeds into a discussion of the ancient pyramids of Egypt. The 3-4-5 relationship is mentioned and the famous rope stretchers are considered in connection with the construction of the pyramids as well as with the beginnings of plane surveying along the Nile. Map and map-making follow with some simple ideas of projective geometry being described through the use of a light source and a translucent sphere.

The film should be usable to aid in introducing geometry although the terminology used may be a bit advanced for a simple introduction.

Some care should be taken by the user to check the understanding of the students' for those words used but not defined in the film.

Arrangements and Combinations. 30 min., $150; el, jh, a.

This film provides a very interesting and clever introduction to combinations. The general theme of the film is to give careful examples of the uses of combinations in problem solving. Some situations are: the various positions of trains of cars, different clothing changes, the number of menus from a few food items, entry of a present-day newspaper puzzle wherein 1,000,000 solutions would have to be entered to guarantee a winner, and several more.

Greatest use of this film is suggested in Grades 7 through 9. The presentation, though very interesting, is probably too frivolous in nature to provide much in the way of mathematical concepts.

How's Chances. 30 min., $150; jh, a.

This film will serve as a good introduction to elementary topics in probability and statistics. It refers to the sampling technique of predicting election outcomes. The binomial and normal distributions are discussed and various applications, such as army supplies and a supermarket, are presented. Motivated by a discussion of an incompleted game of points, the marionettes use simple probability. The film provides excellent motivation for elementary study of probability.

Stretching Imagination. 30 min., $150; jh, a.

Notions of topology are discussed in this film. A fairly intuitive definition of "topologically equivalent" is given and a few examples involving "outside" and "inside" are mentioned. The illustrations include the cup and donut surfaces. Industrial applications are touched upon with items like the roofs of cars and the pressing of plastic materials. A problem involving a paperboy and his shortest delivery route is also mentioned.

The fundamental equation for a polyhedron which expresses the relationship between the number of edges, vertices, and faces is given by an algebraic formula, i.e., $e+2=v+f$. The mobius strip is discussed, as well as the problem of removing one's vest without removing one's coat. This film might be very useful for mathematics clubs at the high school and junior high school levels. The presentation is not given on quite such an elementary level as many of the others in the series.

Sign Language. 30 min., $150; jh, a.

This is an introductory film on trigonometry. The first discussion involves the tangent relation given as a ratio of rise to run. Applications of this ratio are suggested in navigation and artillery. The use of tables is mentioned for computational purposes. Mr. Baird's enunciation of 101 thousandths comes out "101 thousands."

The sine curve is investigated later, although it is presented as disjoint from the preceding discussion of tangent. Much of the vocabulary in the film is used rather glibly, such as the words period, loudness, length and periodicity. Actually, some improper relationships are suggested in this film which make it usable only as an introductory film for a lower level treatment of trigonometry.

Careers in Mathematics. 30 min., $150; jh, sh; tchrs. of el, jh, sh.

This film is a short summary of the eight preceding films and points out that there are careers for women as well as for men in mathematics. Essentially nothing new is presented in this last film of the series and we do not recommend its use separately from the series even for the possible benefit of prospective mathematics students. The film provides a too-brief review of previously discussed topics using references to the earlier films in the series.

ALGEBRA AND POWERS OF TEN. See Special Lessons in Physics.

ALGEBRA OF POINTS AND LINES. See Intermediate Algebra Series.

ALGEBRAIC AND COMPLEX FRACTIONS. See Intermediate Algebra Series.

A PLUS B SQUARED. 1954. sd, bw; 10 min.; jh; International Film Bureau, Inc. $50.

Finding the area of a square motivates the early discussion of $(a+b)^2$. During the presentation the narrator says, "We know how to multiply numbers but how do we multiply letters?" Such language is poor, especially for a film. It is geometrically shown that the area of a square of side-measure $a+b$ is $a^2+2ab+b^2$ but no mention is made of the distributive and commutative laws when $(a+b)^2$ is expanded algebraically. Most teachers could probably present this material as well as it is presented here. In spots the sound was poor on the film used by the reviewers.

ARITHMETIC: ESTIMATING AND CHECKING ANSWERS. 1962. sd, bw; 11 min., el, jh; Coronet Instructional Films, $60.

After an introduction showing the need for estimating answers, procedures for rounding off numbers are presented and applied in illustrated word problems using large numbers and using decimals. Checking the four fundamental processes is stressed.

The type of word problems and the method of illustrating them would appeal to sixth-grade students as an introduction to the topic.

ARRANGEMENTS AND COMBINATIONS. See Adventures in Number and Space.

Axioms in Algebra. 1960. sd, co, 13 min., jh, sh; International Film Bureau Inc., $135.

The axioms discussed in this film are not the axioms of a field but are the traditional version of Euclid's axioms for addition, subtraction, multiplication, division, powers, and roots. The film does not point out that subtraction and division axioms are redundant nor does it restrict the root axiom to principal roots. No recognition of the commutative and associative laws is made. A great deal of effort is made to motivate the learning of each axiom using "practical" illustrations. The narrator speaks of subtracting trucks, adding cages, and multiplying pages which will serve to indicate the lack of consistency with current usage of mathematical language.

Although the photographic techniques are good, this treatment of the axioms in algebra leaves much to be desired.

Base and Place. See Understanding Numbers.

Big Numbers. See Understanding Numbers.

Careers in Mathematics. See Adventures in Number and Space.

Chain Operations. See Engineering Computation Skills: The Slide Rule.

Complex Numbers and Roots of Equations. See Advanced Algebra Series.

Composite Areas. See Engineering Computation skills: The Slide Rule.

Computing Logarithms from Arithmetic and Geometric Series. See Advanced Algebra Series.

Concept of a Function. See McGraw-Hill Teacher Education Series.

Construction of Basic Scales. See Engineering Computation Skills: The Slide Rule.

Conversions. See Engineering Computation Skills: The Slide Rule.

Cosecant, Secant and Cotangent. See Trigonometry Series.

Decimal Numerals. See Junior High Film Series.

Determinants and Cramer's Rule. See Advanced Algebra Series.

Determinants of Any Order. See Advanced Algebra Series.

Developing and Solving Linear Equations. See Intermediate Algebra Series.

Discovering Solids. 5 films, 1959. sd, co or bw; jh, sh, jc; tchrs. of jh. Cenco Ed. Films, Inc., bw $375, co $750.

Solids in the World Around Us. 5 min., bw $75, co $150; jh, sh, jc.

This film shows solid geometric form in everyday life through familiar objects such as flowers, butterfly wings, and even the shell of a turtle. In the manner of an art exhibit, the film shows modern geometric figures such as structural steel, manufactured goods, and rockets. Definitions of point, line, radius, a sector, and other terms are given near the conclusion of the film.

The use of natural and artificial art is very effective and will make the film quite useful for motivation in geometry. Although the film contains little explaining, it is definitely a worthwhile and enjoyable film.

Volumes of Cubes, Prisms, Cylinders. 5 min., bw $75, co $150; jh, sh; tchrs. of jh.

After an examination of solids that exist in the world around us, a cube is defined to be the basic "unit" used in the calculation of volumes. Rectangular prisms are seen to be made up of a series of cubes. Various odd shaped figures are made and their volumes computed by counting the number of cubic units from which they were constructed. An excellent development of the formula for the volume of a rectangular prism follows. The formula for the volume of a cylinder is developed from the relationship already established for the prism.

The development of these formulas is excellent and easily followed. The illustrations and practical applications are well chosen.

Volumes of Pyramids, Cones, and Cylinders. 15 min., bw $75, co $150; jh; tchrs. of jh.

Volume is introduced as it is seen in everyday life with animation being used to illustrate that the volume of a prism is equal to area of base times height. The formula for the volume of a pyramid is thoroughly illustrated and leads to the formula for the volume of a cone. Finally, in the same careful manner, the formula for the volume of a sphere is derived.

Animation is used to excellent advantage and the film is definitely a "must" for almost any group studying volume.

Surface Areas of Solids, Parts I and II. 15 min. each, bw $75, co $150; jh, sh; tchrs. of jh.

In the first film, surface areas of cubes, prisms, and pyramids are considered, drawing applications from real life situations. The second film considers surfaces of cylinders, cones, and spheres which are shown to be surfaces of revolution. In both films, the formulas are developed through animation and applied to everyday situations.

While the development of the formulas is well done, those on surfaces of revolution will require additional amplification by the teacher. In any case both films will make excellent introductions to the topics.

DONALD IN MATHMAGIC LAND. 1959. sd, co, 26 min., el, jh, sh, jc; tchrs. of el, jh, sh; Walt Disney Productions, $250 (10 yr. lease).

Donald Duck enters a fantasy land of animated numerals and geometric forms and is guided in his journey by a "Spirit of Adventure" whose articulation is much clearer than Donald's. The relation between length and pitch of a vibrating string serves as an excuse for a visit to a Pythagorean jam session. Animated diagrams of the Pythagorean's symbol, the pentagram, show its connection with the golden rectangle and the occurrence of these forms in art and nature. After a discussion of games, including some views of expert billiard shots, the scene shifts to a miscellany of plane and solid geometric forms and their physical applications. Repeating pentagrams within pentagrams and other similar situations are used to introduce the notion of infinity.

The viewer will be attracted by the music, art, animation, and humor rather than by the mathematics. There is probably some motivational value here but an extensive follow-up will be required in order to teach mathematical concepts. Nevertheless, the film is good entertainment for almost any age level.

DOUBLE AND HALF ANGLE FORMULAS. See Trigonometry Series.

EARLIEST NUMBERS. See Understanding Numbers.

EFFICIENT OPERATIONS I. See Engineering Computation Skills: The Slide Rule.

EFFICIENT OPERATIONS II. See Engineering Computation Skills: The Slide Rule.

EIGHT FUNDAMENTAL TRIGONOMETRIC IDENTITIES. See Trigonometry Series.

ELECTRONIC COMPUTERS AND MATHEMATICS. 1961. sd, co, 25 min.; jh, sh, jc, a; tchrs. of jh, sh; bw $110, co $220.

The history of computers is shown in this film from finger counting, use of pebbles, the abacus, to modern electronic giants adaptable to many purposes. The binary system is explained and compared with the decimal system. Many illustrations of working computers are given, identifying the major components such as input, storage, processing, and output units.

Although the film is very interesting and stimulating, the emphasis is on the mechanical and vocational aspects of computers rather than the mathematical aspects. This does not detract from its use as a film for motivation.

ELEMENTS OF TRIGONOMETRY. See Special Lessons in Physics.

ENGINEERING COMPUTATION SKILLS: THE SLIDE RULE. 15 films, 1960. sd, bw; sh, jc, a; Bureau of Audio-Visual Instruction, State University of Iowa, $1,025; guide, manual.

The series is best suited for use in service courses designed primarily for pre-engineering or science areas where interest is in the manipulative or technical skill in the use of the slide rule. There are no mathematical notions derived or developed with any degree of rigor in any part of the series. We feel that the presentation is well organized, that the lecturer is well prepared, and that his use of props is generally quite adequate.

The mathematics is basically correct. There are some misuses of terminology, and a question concerning the cancellation of units arises in one or two films of the series. The lecture and demonstration methods of instruction are appropriate for the subject matter and seem to be paced so that the average high school or college student should be able to follow along easily. The devices used for instruction are very good and are not generally found in the classroom, especially the chalkboard faced slide rule used in demonstrating the construction of scales. Most of the films end with a brief but thorough review of the topics discussed in that film.

We recommend the series as very useful in teaching skill in the use of the slide rule. The first two or three films could be used separately but the remainder of the series should be used in a consecutive sequence.

Construction of Basic Scales. 23 min., $75; sh, jc, a.

The construction of the C, D, and A scale is very carefully done. A chalkboard faced slide rule is skillfully used in the film. This film might be used separately from the remainder of the series whenever an explanation of the means of construction of the simple slide rule is desired. Possibilities for use by mathematics clubs or for enrichment programs also exist.

Multiplication and Division. 30 min., $75; sh, jc, a.

This film shows the two standard demonstration slide rules of the series with the CI, CF, DF, and CIF scales. The construction of these scales and their relation to the C and D scales are briefly noted. The film does a good job of exhibiting the slide rule as a tool and showing its application to problems in mathematics. It can be used separately from the remainder of the series.

Chain Operations. 28 min., $75; sh, jc, a.

In this film the lecturer uses the entire family of the C and D scales to solve some simple problems in multiplication and division.

We do not recommend the use of this film separately from the previous film because the identification of the scales is made in the earlier film. The procedure and treatment is quite formal, although the devices are excellent. The positioning of the decimal point is done by an approximation procedure rather than a technique involving scientific notation.

Ratio Problems. 29 min., $75; sh, jc, a.

By solving problems involving ratio, the lecturer gives a good example of the finesse and ingenuity required of the skilled slide rule operator. He insists on the requirement that both the hairline and slide must be moved no more than once in a given operation, hence calling for some ingenuity on the part of the operator. The presentation then proceeds to show a single setting method involving the C and D scales to solve problems involving proportionality.

Conversions. 30 min., $75; sd, bw; sh, jc, a.

The question of converting from one system of units to another is discussed. The various uses of equality have not been discussed in any of the previous films; however, in this film, the lecturer makes a distinct point of "1 ft. ≠ 12 in." The reason given for such ~~a statement~~ is that "1 ≠ 12" and "ft. ≠ in." The need for this particular illustration is not clear and as a convenience, portions of this notion are discarded later. We recommend that the concept and technique of "cancelling units" be thoroughly discussed with the intended viewers before this film is shown.

Squares, Cubes, and Roots. 30 min., $75; sd, bw; sh, jc, a.

This film contains an adequate presentation of the technique of use and application of the A, B, K, R_1, and R_2 scales. Simple illustrations are used to find the second and third powers and second and third roots of numbers.

Efficient Operations I. 27 min., $75; sd, bw; sh, jc, a.

The theme of this film is the use of shortcuts in using the slide rule. The film opens with a discussion of the importance of the study of logarithms. In this discussion the lecturer refers to a set of examples on the chalkboard which are too numerous for easy comprehension and seem contrived. We feel that this discussion should have been placed much earlier in the series since the notion of logarithm is at the foundation of the theory behind the slide rule.

Manipulations of the slide rule to solve equations in the form $y = kx^2$ are demonstrated for slide rules with A and B or R_1 and R_2 scales. The manipulation necessary for finding the area of a circle in one setting is shown, as well as a method of finding the circumference of a circle.

Efficient Operations II. 32 min., $75; sd, bw; sh, jc, a.

The content of this film is similar to the previous one, and the theme, namely that of short cuts in the use of the slide rule, continues. The topics considered range from area comparisons to approximate solutions of a quadratic equation. The latter topic is restricted to those quadratic equations where one root is known or, at least, can be estimated. In the summary, the comment is made that the intent was to obtain solutions to problems not necessarily directly adapted to the slide rule.

Raising Numbers to Powers. 31 min., $75; sd, bw; sh, jc, a.

The film deals with the construction and use of log log scales discussed to an extended degree. When factoring an expression such as 8^5 to obtain $8^3 \cdot 8^2$, the lecturer calls this "factoring the exponent" which seems to us to be inappropriate. The same phrase is used to describe the sentences: $8^5 = (8^{2.5})^2$ and $8^5 = 2^5 \cdot 4^5$. In the summary at the conclusion of the film, the lecturer uses what we feel is more appropriate terminology in dealing with this issue.

Roots and Exponential Equations. 29 min., $75; sd, bw; sh, jc, a.

The film begins with a good review of the previous film and then proceeds with the general problem of extracting roots by means of the slide rule. The symbol $N^{1/r}$ is written on the chalkboard on several occasions and the lecturer leaves us with the impression that r is the root of N rather than the index of the root. We are also left with the feeling that the extraction of roots is simply a mechanical process since no mention is made of the theoretical background involved. The relation between the log log scales and the natural logarithm of a number is not made clear in this film although an example is given involving this relationship.

Trigonometric Scales. 31 min., $75; sd, bw; sh, jc, a.

An introduction to the use of the slide rule in trigonometry is given in this film. The trigonometric scales of the slide rule are constructed with care and in a manner which is descriptive of the nature of the sine and cosine functions. Graphs of the sine function and log sine function for small arguments are shown and are quite adequate for the purpose of the film.

Decimal notation for degrees and fractions thereof are justified on the basis that they are more useful in engineering and science, and the scales on the demonstration slide rule are labeled accordingly. The cofunction relationship between sine and cosine is described in detail and is used to justify the use of the S scale to find values of the cosine function. The T and ST scales are constructed and discussed.

Little computation is performed in this film and consequently it moves rather rapidly. This film might be used as an introduction to the use of the slide rule in trigonometry by giving an explanation of the construction of the trigonometric scales.

Right Triangles. 32 min., $75; sd, bw; sh, jc, a

Methods of solving right triangles are discussed with emphasis on the problem: given two sides, find the hypotenuse and angles. The sine-tangent method, which solves the problem in two settings, and an approximation method, are derived and demonstrated.

The following quotation, taken directly from the sound track, serves to illustrate the degree of detail used in teaching the techniques of slide rule manipulation as well as to illustrate other features of the series. "... and thirdly we have the relationship that says the angles of a right triangle are complementary. To handle these problems, we often use our sine-tangent technique that is particularly useful when we know the values of the two sides of a triangle and are looking for the hypotenuse. This method we know to be good for all cases of angle A greater than 5.75°. If the angle is less than 5.75°, we *must* use our approximation method. In the approximation method itself, which is: the hypotenuse is equal to the long side plus the short side squared over twice the long side, is good up to 23°. So we see the sine-tangent method will work down to 5.75° and the approximation method up to 23°. These methods, then, will help us solve for right triangles using our slide rule."

Right Triangle Applications. 31 min., $75; sd, bw; sh, jc, a.

The film begins with a synopsis of the previous film and the precise steps for calculation in the sine-tangent method are listed. A few very simple problems are worked out using this method from definitions of the cosine function as equal to the "adj" over the "hypot" and sine function as equal to the "opp" over the "hypot."

Mention is made of types of problems involved in practical applications but no example from any of the types is worked out in detail. There are no significant examples of any sort completed in this film.

Scalars and vectors are used as different concepts but no clear distinction is made between them and no definition of vector notation is given.

Oblique Triangles. 30 min., $75; sd, bw; sh, jc, a.

A chart is used to coordinate the relationships for the general triangle. The sum of the angles of a triangle is given as 180°; the Sine and Cosine Laws are stated. Fairly trivial geometric illustrations of each of these are presented using the slide rule whenever possible for calculations. We note the absence of any reference to the Law of Tangents. Some of the problems dealt with could be more easily solved with the slide rule by using the Law of Tangents.

There is an unusual amount of very distracting busywork on the chalkboard at the end of the film.

Composite Areas. 26 min., $75; sd, bw; sh, jc, a.

The computation of areas of polygons and curves by considering them as composite areas is discussed. The composite area is determined by summing the areas of triangles, rectangles, segments, and sectors. The slide rule is used whenever possible to calculate these partial areas. The constructions and diagrams used for dismemberment are very well executed. No summary of the series is included at the end of this final film.

ENGINEERING PROBLEMS. 6 films, 1958. sd, bw; sh, jc; Bureau of Audio-Visual Instruction, State Univ. of Iowa, $400.

The use of the series is recommended only in situations where qualified staff is not available to teach logarithms or to provide a service course for the techniques of logarithmic manipulation. The films could serve only in a situation wherein the student has no demand for mathematical theory.

Nothing is done in this series which could not be done better in a classroom by most regular faculty members. The lecture method is used throughout the entire series with the result that the presentation is very formal and, in turn, very uninteresting. In the early films the lecturer often mumbles while performing computations at the chalkboard. The lecturer habitually speaks directly to the chalkboard with no microphone provided for such action making it difficult to hear what is said. Generally, the use of the chalkboard is good, and the solutions of examples are presented in step-by-step developments rather than referring to pre-obtained results. The camera techniques are also to be commended although some drawings and sketches are smaller than desirable and make for insurmountable problems for the cameraman. This is especially true of the film on interpolation of mantissas.

The mathematical concepts presented are basically correct, although not generally couched in contemporary terminology. Exceptions to this are the derivations and definitions of the trigonometric functions which are done very well in the manner of circular functions. No applications of logarithms to any mathematics or science are presented. There are many instances of poor phrasing and symbolism. In this category we include the mix up between number and numeral, the omission of identifying symbols such as degree and minute marks and decimals before mantissas, the consistant substitution of the word "power" for "exponent," and the phrasing which leads the viewer to believe in the exactness of decimal approximations to irrational numbers, e.g., $\log_{10}e = .4343$.

A student may achieve some understanding of techniques of logarithmic manipulation and their use in problem solving through the use of these films. However, we suggest that there will be serious confusion about certain mathematical concepts for those students who may be intertested in pursuing careers in mathematics rather than careers in engineering or technical fields.

Logarithms—Characteristics. 26 min., $75; sh, jc, a.

We cite the following errors: the logarithm of one to the base two is denoted as two; the first power of ten is said to be equal to one. These errors indicate a need for editing. We also call attention to the need for consistency in "rounding off numbers."

In this film there is a mix up between "number" and "numeral" which is further complicated by the use of such phrases as "behind the number" and "the plus part of a number." Such statements as "The power of a number is a number," and "In scientific notation, the power of ten is the characteristic of the number," exemplify that "exponent" and "power" are often interchanged.

The illustrative examples seem contrived; they consist of the ordered sequence of the digits 1, 2, 3, 4, 5, 6, and 7 with the decimal point being placed in various positions, e.g., 123.4567, 1234.567, 12.34567, etc. A more random sequence of digits would have been more effective and illustrative.

Mantissas. 26 min., $75; sd, bw; sh, jc, a.

The film deals with obtaining the mantissas of logarithms. It gives a good graphic portrayal of linear interpolation to find an approximation to a desired mantissa. The non-linearity of the logarithm function is pointed out as is the partial correction of interpolation through the use of larger tables. We feel it is poor pedagogy to omit symbols for degrees and minutes and decimal points before mantissas.

Logarithmic Operations I, II. 27 min., each, $75; sh, jc, a.

In these films, the algebraic theorems involving exponents are applied to the definition of logarithm to arrive at the theorems on logarithmic operations. The illustrative examples, however, seem to be contrived and have little meaning to the viewer. The lecturer seems to be mumbling to the chalkboard a good deal to the extent that he appears to be talking to himself.

These films follow the mood of the series by teaching the manipulation of logarithms rather than theory of logarithms. The objective, therefore, seems to be to develop the tool of logarithmic manipulation.

In the second film of this pair, log logs are investigated as well as fractional powers of fractional numbers. The definitions of the trigonometric functions are given by means of coordinate geometry. A table of algebraic signs is constructed for the four most important trigonometric functions and is to be memorized. This again emphasizes the stress being placed in this series on manipulative rules and memorized devices.

Trigonometric Applications. 32 min., $75; sh, jc, a.

Contained in this film are a brief review of the unit circle definition of the trigonometric functions, statements of the Law of Sines and Law of Cosines without proof of either, and examples of the application of each of these laws. Finally, definitions of sin, cos, and tan are given in terms of a right triangle. The lecture method is continued and the presentation is extremely formal.

Logarithmic Systems. 21 min., $75; sh, jc, a.

The last of this series compares logarithms of base ten with logarithms of base e. Base e is tied to base ten in such a specific manner that the viewer may be left with the impression that logarithms of base e exist only as related to logarithms of base ten. The derivation of characteristics and mantissas for base e could have been given just as formally as those for base ten and without the use of logarithms of base ten. Unfortunately, the viewer may also have the impression that the logarithm of e to the base ten is exactly 0.4343.

EQUATIONS AND GRAPHS OF THE PARABOLA. See Intermediate Algebra Series.

EQUATIONS WITH UNKNOWNS IN THE EXPONENTS. See Intermediate Algebra Series.

FIVE FUNDAMENTAL POSTULATES OF ALGEBRA. See Advanced Algebra Series.

FORMULAS IN MATHEMATICS. 1960. sd, co, 10 min.; jh; International Film Bureau, Inc., $110.

A treatment is given of the distance formula $D=rt$ with the example of an airplane in flight. The use of this formula is illustrated.

A perplexing use of language is noticed in this film. For example, the narrator states that ninety miles divided by one hour is ninety miles per hour. This will likely make it seem as if division is an operation on objects other than numbers. The viewer is cautioned to use the same units of measure but is not told why. We find it difficult to conceive of a place in a mathematics program where this film might be profitably used.

FRACTIONS. See Understanding Numbers.

FUNDAMENTAL OPERATIONS. See Understanding Numbers.

GENERAL METHODS FOR SOLVING QUADRATIC EQUATIONS. See Intermediate Algebra Series.

GRAPHING LINEAR EQUATIONS. 1961, sd, co, bw, 12 min.; jh, sh; Coronet Instructional Films, bw $60, co $110, guide.

The step-by-step construction of linear graphs is shown in detail as well as the relations between the line graph and the points in rectangular coordinates. Slopes and intercepts are discussed as well as the effect that changes in these parameters have on the graph.

In general the film is good and would be useful for senior high school algebra students but the discussion of slope and intercept will need extension.

GRAPHS OF PERIODIC FUNCTIONS. See Trigonometry Series.

GRAPHS OF QUADRATIC EQUATIONS. See Advanced Algebra Series.

HISTORICAL INTRODUCTION TO ALGEBRA. See Advanced Algebra Series.

HOW MAN LEARNED TO COUNT. See Adventures in Number and Space.

HOW'S CHANCES. See Adventures in Number and Space.

HYPERBOLA, ELLIPSE AND CIRCLE. See Intermediate Algebra Series.

THE IDEA OF NUMBERS: AN INTRODUCTION TO NUMBER SYSTEMS. 1960. sd, co, 14 min.; el, jh; tchrs. of el, jh; International Film Bureau, Inc., $135.

The development of the number concept is given historically, but confusion arises in this film as to the difference between number and numeral. Systems of numeration such as the Babylonian, Mayan, Arabic, and Roman are discussed as well as calculating aids that have been used. Place value systems are noted with some emphasis on base two. The notion that a number is an idea, not a mark on paper, is not made clear and is further complicated by terms like "three place number." However, with careful correction of these errors by the teacher using this film, we recommend its use because of its technical qualities and its historical material.

IMAGINARY AND COMPLEX NUMBER. See Intermediate Algebra Series.

INFINITE SERIES AND THE BINOMIAL EXPANSION. See Intermediate Algebra Series.

THE INTEGERS. See Junior High Film Series.

INTERMEDIATE ALGEBRA SERIES. 24 films, 1959. sd, bw; Modern Learning Aids, $3,600; sh, jc; sh.

This series of twenty-four films presents a selection of topics usually included in a second course in high school algebra.

Natural Numbers, Integers and Rational Numbers. 30 min., $150; jh, sh, jc; tchrs. of jh, sh.

The instructor considers the natural numbers and some of their properties. These include comparison of magnitudes, addition, and multiplication along with the allied notions of closure, identity elements, and the commutative, associative and distributive laws. Solution of simple equations is discussed, but the viewer may be left with the impression that $-a$ is a negative number. Reciprocals are used to introduce the rational numbers.

Despite a few misleading examples, the film is usable for algebra classes or for in-service teachers.

Addition and Subtraction of Rational Numbers. 30 min., $150; sh, jc; tchrs. of sh.

After a brief review of the preceding film, the instructor proceeds to a consideration of the number line. It is indicated that every rational number corresponds to a point on the line and the absolute value of a number is defined to be the distance between a point corresponding to the number and the point corresponding to zero.

Although some proofs of theorems involving additive inverses are given, no mention is made of the dependence upon the uniqueness of the additive inverse. The "rules" for addition in terms of absolute value are poorly presented. The convention in connection with the equality of $3-2$ and $3+(-2)$ is not explained.

The content of this film is not as well presented as in the first film, nor is it presented in the spirit of contemporary mathematics.

Multiplication of Rational Numbers. 30 min., $150; sh, jc; tchrs. of sh.

Theorems concerning multiplication of rational numbers are given, but the basic theorem on the uniqueness of the additive inverse is again slighted. The viewer may also be led to believe that $(-a)(-b)=ab$ is a theorem concerning the product of two negative numbers since the symbol $-a$ is not clearly defined. Division is covered hastily and includes a discussion of the three signs of a fraction, which is not the best way of handling this. Algebraic expressions are considered and the division of polynomials is discussed in a strictly traditional fashion.

Developing and Solving Linear Equations. 30 min., $150; sh, jc; tchrs. of sh.

The instructor discusses the solving of linear equations with restrictions placed on the variable x. Graphs are used to picture the set of points which satisfy these restrictions. The instructor is not careful in his use of the language when he states "we want x alone" and "'x' is in the parentheses and clearly we must get it out." The checking of a solution is emphasized.

The film could be used to introduce linear equations or as a quick review but should not be used in contemporary mathematics programs.

Solving Simultaneous Linear Equations. 30 min., $150; sh, jc; tchrs. of sh.

After a brief review of the solving of linear equations in the variable x, conditions involving two variables are discussed by means of examples. The domains of the variables are not specified. The procedure for picturing an ordered pair of numbers is outlined.

Although a linear function is graphed, no mention is made of the terms "intercept" or "slope." Graphing is illustrated as an approximate method of finding a simultaneous solution of two linear conditions by inspecting the pictures of the graphs. A pair of simultaneous equations is solved algebraically. Nothing is said of the concept of equivalent equations. The solving of simultaneous conditions involving inequalities is not considered.

More Solutions of Linear Equations. 30 min., $150; sh, jc; tchrs. of sh.

This film deals with the problem of solving verbal problems. Four such problems are worked as examples. These involve a problem on digits, mixtures, and uniform velocities. It is clearly indicated that the solution to the equation is a number while the solution to the problem may be a number of units. The viewer is assured that there is no set procedure in the solution of verbal problems.

This film is very good and could make a valuable contribution to any course in intermediate algebra.

Special Products and Factoring. 30 min., $150; sh, jc; tchrs. of sh.

Although the instructor uses the word "factor" in explaining the process of "factoring," the distributive law is used as related to factoring a common factor and to the multiplication of binomials. Little is said of the basic use of factoring, that is, to write equivalent expressions in a different form, and too much is said about the time worn gimmicks for factoring the usual types of quadratics.

In discussing a^3-b^3, the instructor relies too heavily upon the experience of the student to suggest the factor $a-b$. More needs to be said about the reasons why a^2+b^2 is not factorable. The instructor evidently assumes that a, b, c, and d are integers only when finding, by trial and error, factors of the form $(ax+b)$ and $(cx+d)$. The domain of these variables should be extended to the rational numbers, real numbers, and possibly even the complex numbers for intermediate algebra students. Another error is indicated by the statement that $\sqrt{a^2}=a$, which should be restricted to non-negative numbers.

This film certainly needs much supplementation if it is to be used at any level.

Quadratic Equations. 30 min., $150; sh, jc; tchrs. of sh.

The discussion of $ax^2+bx+c=0$ as a quadratic equation does not restrict a to non-zero numbers. Use is made of the statement, "If $a \cdot b=0$, then $a=0$ or $b=0$," but should be clearly stated as a bi-conditional statement. The method of completing the square is handled well by using the statement, "$x^2=y$ if and only if $x=\pm\sqrt{y}$."

However, the method of completing the square is not shown to be a method of factoring over the real numbers and inequalities are not developed parallel to this work with equalities. More work will be needed with the method of completing the square in order to make the derivation of the quadratic formula meaningful to most students.

The film could be used to summarize the factoring and completing the square methods of solving quadratic equations if the curriculum design is along traditional lines.

Algebraic and Complex Fractions. 30 min., $150; sh, jc; tchrs. of sh.

Without proof the instructor states that the reciprocal of a product is the product of the reciprocals and assumes the uniqueness of the multiplicative inverse. No complete proof of

$$\frac{a}{b}\times\frac{c}{d}=\frac{ac}{bd}$$

$b\neq 0$, $d\neq 0$, is given and the instructor "obtains the value" of

$$\frac{\frac{a}{b}}{\frac{c}{d}}$$

instead of obtaining an equivalent expression. Use is made of the commutative and distributive laws but no mention is made of them when they are used. Cancelling is described in terms of dividing the numerator and denominator by the same number without mentioning restrictions. The summary is a restatement of traditional rules of manipulation which serves to indicate that the function of the film will be as a lesson in symbol manipulation.

Because the basic axioms developed in the first film of the series are not emphasized here, the film may be of questionable value to algebra teachers who are stressing the structure of algebra.

Solving Equations in Fractional Form. 30 min., $150; sh, jc; tchrs. of sh.

A fractional equation is defined as one in which the unknown appears in a denominator. It is noted that it immediately follows that $x\neq 0$ since division by zero is not defined. Examples are used to illustrate technique which is set down by three rules:

1. Multiply by the least common denominator.
2. Solve the resulting equation.
3. Check the solutions.

The use of such a list of steps is questionable since it leaves no room for originality and, in fact, the instructor does not always conform to these steps himself.

In general, the instructor does a good job of handling traditional material.

Algebra of Points and Lines. 30 min., $150; sh, jc; tchrs. of sh.

The instructor uses the background of the development in the fifth film of this series, "Solving Simultaneous Linear Equations," and begins by asking what the equations of parallel lines have in common. Slope and y-intercept are then discussed. Slope is defined in terms of increments and not defined for lines parallel to the y-axis. It is not made clear whether the term y-intercept refers to the number b or the point $(0,b)$. The equation of a line through two points is discussed in the conclusion of the film.

The instructor does a good job of presenting this material but the film covers too much ground to be used for any purpose other than review.

Variation: A Lesson in Reading. 30 min., $150; sh, jc; tchrs. of sh.

The film emphasizes the many verbal expressions we have for the functions defined by equations of the form $y=kx$ and $y=\frac{k}{x}$. Of particular interest is a definition of inverse variation in terms of direct variation, namely: y varies inversely as x if and only if y varies directly as the reciprocal of x.

Examples from geometry and physics are used to illustrate variation problems. Quadratic and cubic variation are discussed as well as linear variation and joint variation.

This film could be used not only in algebra and physical science classes but also in teacher training classes as an example of sound mathematics teaching.

Radicals and the Real Number System. 30 min., $150; sh, jc; tchrs. of sh.

After a brief review of earlier comments on irrational numbers, a short historical sketch of the discovery of non-rational numbers, and a sketch of $\sqrt{2}$ as the diagonal of a square of unit side, the instructor extends the numeration system by defining a number for which he has a need and which is not in the present system. This was well done and included a brief comment on imaginary numbers.

Addition of radicals is presented and associated with the distributive law and the rules for multiplying and dividing square roots. The instructor emphasizes an often overlooked notion that $\sqrt{3}$ has only a decimal approximation and no decimal equivalent.

Many additions will need to be made by the teacher who uses this film. The fact that $\sqrt{x^2}=-x$, if $x<0$, should be stressed, as well as the fact that the operations with real numbers are the means to write expressions in other equivalent forms.

Use of this film is suggested for viewing as a summary of material which has already been discussed in class.

Roots of Higher Order. 30 min., $150; sh, jc; tchrs. of sh.

A recursive definition of radicals is given and product and quotient of radicals is extended to orders higher than two. Imaginary numbers are introduced as well as complex numbers and their notation.

The film is quite carefully done and could be used by students as a summary for material covered in class.

Imaginary and Complex Numbers. 30 min., $150; sh, jc; tchrs. of sh.

Complex numbers are defined as numbers of the form $a+bi$ where a and b are real numbers and $i^2=-1$. Addition and multiplication of complex numbers are illustrated.

All the mechanical aspects of complex numbers are included in this film including a brief discussion of complex numbers plotted on a plane but the really important ideas are missing. Nothing is said about the ultimate objective which is to construct a new system of numbers which will contain the real numbers as a subset, which will have all the properties of the real numbers, and which will contain solutions for equations which have no solution in the real numbers. The construction of the system of complex numbers as ordered pairs of real numbers is not hinted at here.

It is hard to see how this film will be of any help in clarifying the introduction to complex numbers.

Working with Positive and Negative Exponents. 30 min., $150; sh, jc; tchrs. of sh.

Integral exponents are defined together with the rules of operation for positive integral exponents. The discussion brings out the rationale behind the definition of negative and zero exponents. A discussion of scientific notation brings out the subject of significant digits.

The film could be used as an introduction to or review of integral exponents.

Using Fractional and Rational Exponents. 28 min., $150; sh, jc; tchrs. of sh.

Definitions of fractional exponents are given and shown to have properties consistent with the properties of integral exponents. Some further discussion of the restrictions on the base will be called for from the teacher. Examples show the use of fractional exponents in treating roots of various orders.

General Methods for Solving Quadratic Equations. 30 min., $150; sh, jc; tchrs. of sh.

A quadratic equation in the variable x is defined as a condition which can be put in the form $ax^2+bx+c=0$ where $a\neq 0$ and a, b, and c are real numbers. It is pointed out that the parameters could be complex numbers but that the discussion here will be limited to reals.

The methods of factoring and completing the square are reviewed and the quadratic formula is derived. Relationships between the parameters and the sum and product of roots are given, and the use of the discriminent to determine the character of the roots is outlined.

In general this film is well worth showing to intermediate algebra students.

Equations and Graphs of the Parabola. 30 min., $150; sh, jc; tchrs of sh.

Graphs of equations of the form $y=ax^2+bx+c$ and close relatives are considered in this film. The restriction $a\neq 0$ is not mentioned. Pictures of the various graphs are compared to indicate the effects of changes in parameters. Axis of symmetry is discussed in relation to the process of completing the square, and the term vertex is defined.

The film's best use will be in reviewing the parabola.

Hyperbola, Ellipse and Circle. 30 min., $150; sh, jc; tchrs. of sh.

A brief review of the parabola is presented but the necessary restriction $a\neq 0$ is omitted. The point is made that one draws a picture of a graph and does not draw the graph itself. Two general types of hyperbolas are indicated, namely: $xy=c$, and $ax^2+by^2=c$ $(ab<0)$. Discontinuity is not mentioned and asymptotes need fuller explanation. Ellipses are discussed and the circle is seen as a special case of the ellipse. Solutions of simultaneous linear and quadratic equations are given by graphing and algebraic techniques. Finally, solutions of two simultaneous quadratic equations are considered and the methods of solution are outlined.

Although this film gives a good presentation of these topics, the competent teacher could do as well in the regular classroom.

Progressions, Sequences and Series. 30 min., $150; sh, jc; tchrs. of sh.

Arithmetic and geometric progressions are discussed including the four elements of an arithmetic progression, namely: first term, common difference, number of terms, and last term; similarly, the elements of a geometric progression are discussed. A series is defined as the sum of the terms in a progression. Analytic methods for determining the sum of terms of arithmetic and geometric progressions are discussed fully.

The words "progression" and "series" are confused on occasion. Grouping symbols are badly needed in the formula for the sum of a geometric progression. The development is much too rapid for an introduction to these ideas. Not enough examples are given for either progressions or series.

Infinite Series and the Binomial Expansion. 29 min., $150; sh, jc; tchrs. of sh.

Using a geometric series in which the number of terms added increases without bound, it is demonstrated that if the common ratio of a geometric series falls in the range from -1 to 1 and if the number of terms increases without bound, then the sum, $S_\infty=a/(1-r)$. A discussion of limits and repeating decimals follows. The film concludes with a statement concerning the binomial expansion and the application of the binomial expansion to ar example.

The instructor fails to define terms used in the discussion. Series has never been carefully defined here. The development of the rationale behind the coefficients in the binomial expansion is entirely omitted. We fear that the viewer may gain the impression that getting a rule or formula is the heart of mathematics. The nature of limits is poorly handled and no mention is made of $r=0$ in the sum of an infinite geometric series.

Equations with Unknowns in the Exponents. 30 min., $150; sh, jc; tchrs of sh.

A general exponential equation is given, namely: $b^x=a$. It is mentioned that this equation has a solution if b is positive and not equal to one and if a is positive. It is shown that $b^x=a$ is equivalent to $\log_b a=x$. Basic ideas of logarithms are then discussed along with the use of scientific notation. Definitions of mantissa and characteristics are given.

In the equation $b^x=a$, no mention is made as to why such an x exists and none as to why $b^x=b^y$ implies $x=y$. The property that log $(ac)=\log a+\log c$ is given without any explanation. The treatment of scientific notation is poorly done.

Using Logarithms to Solve Equations. 30 min., $150; sh, jc; tchrs. of sh.

Beginning with $5^x=3$, the fact that $x=\log 3/\log 5$ is derived. This indicates a need for tables of approximations. These tables could be used to find approximate values for products, quotients, and roots of numbers.

The lecturer gives us the impression that each rational number can be written with a finite number of decimal places. "Distance" between numbers is mentioned, and the distinction between ratio and proportion is not made clear. It is pointed out that solutions with logarithms are approximations, and the emphasis on checking the reasonableness of solutions is good.

INTERPOLATION IN TRIGONOMETRIC TABLES. See Trigonometry Series.

INTRODUCTION TO FACTORING. See Advanced Algebra Series.

INTRODUCTION TO GRAPHS OF EQUATIONS. See Advanced Algebra Series.

INTRODUCTION TO LOGARITHMS. See Trigonometry Series.

INTRODUCTION TO QUADRATIC EQUATIONS. See Advanced Algebra Series.

INTRODUCTION TO SIMULTANEOUS EQUATIONS. See Advanced Algebra Series.

IRRATIONAL NUMBERS. See McGraw-Hill Teacher Education Series.

JUNIOR HIGH FILM SERIES. 6 films, 1962. sd, bw; jh, tchrs. of jh; Educational Research Council of Greater Cleveland, \$720; guide.

Numeration Systems. 30 min., \$720 the set; jh; tchrs. of el, jh; guide.

The film begins with remarks on the positional nature of our numeration system based on ten. Supposing that seven had been originally chosen as a base instead of ten, the lecturer does some counting in base seven and goes on to discuss the role of seven and its powers, addition and multiplication in base seven, and carries out some manipulations. Binary numeration is discussed, including tables of addition and multiplication. The film closes with some implications resulting from choice of greater and lesser bases.

The mathematics in this film is without fault, and the presentation has been done beautifully in a concise and informative manner. The film will find use in junior high school classes and would be valuable for teacher training and in-service work.

The Whole Numbers. 30 min., \$720 the set; jh, sh; tchrs. of el, jh, sh; guide.

The film deals mainly with the whole numbers and emphasizes the difference between numbers and the symbols which represent them. Commutative, associative, and distributive principles, as well as properties of 0 and 1 are discussed. The instructor illustrates the fact that by applying these principles we can cut down on the number of facts that need to be learned.

Elementary teachers might profit from the discussion of computation which shows how the steps involved in the process can be justified. The film concludes with some comments on the association of arithmetic and geometry through the number line.

The film could be used as an introduction to structure for junior and senior high school students and as a teacher training film.

The Integers. 30 min., \$720 the set; jh, sh; tchrs. of jh. sh; guide.

Basically, the film demonstrates a fairly rigorous introduction to negative integers. The need for signed numbers is suggested by the notion of a thermometer with degrees above and below zero. The existence of negative numbers is postulated and it is assumed that the unique solution of $a+b=0$ is $b=(-a)$. After a brief digression to define absolute value, operations with the integers are illustrated. Proofs are based on fundamental postulates.

The pace of this film makes it more suitable for teacher training than for viewing by students.

The Rational Numbers. 30 min., \$720 the set; jh; tchrs. of el, jh, sh; guide.

It is seen that measurement necessitates numbers in addition to the set of integers. The number called "a divided by b" is defined as the number such that $(a \div b) \cdot b = a$ for all integers a and b except $b=0$. This set of numbers is then named the set of rational numbers and it is seen that the integers form a subset of the rational numbers. Using the basic laws of integers, the properties of rational numbers are derived.

Although the development does not follow a completely rigorous design, the subject is treated quite well and may, in fact, give many teachers a clearer picture of the rational number system. The students will probably find the film most satisfactory as a review.

Dec. ıl Numerals. 30 min., \$720 the set; jh, sh, tchrs. of el, jh, sh; guide.

The positional numeration, base ten, is extended to include not only numerals for whole numbers but also all rational numbers. This is accomplished through the process of division. Repeating and non-repeating decimals are noted making the point that if $\frac{a}{b}$ is a rational number, then $\frac{a}{b}$ has a decimal numeral which either terminates or is periodic. The converse of the preceding conditional is also indicated as being true. The teacher's film guide will be of great help in using this film.

We heartily recommend the film for viewing by all teachers of elementary and secondary mathematics. Although it proceeds at a rather rapid rate, it would be suitable for pupil viewing where the student has been well prepared and especially suited for review. The film provided the reviewers had poor sound and synchronization in spots.

Language of Algebra. 30 min., \$720 the set; jh, sh; tchrs. of jh, sh; guide.

Teachers who feel that the modern approach to algebra is not for them should view this film. The lecturer gives a concise introduction to algebra which is very illuminating. Open sentences, set selector, and solution set are discussed as well as variables and domain.

The film will be excellent not only for teachers, but also as an introduction to algebra for beginning students.

LANGUAGE OF ALGEBRA. 1960. sd, co, 16 min., jh, sh; International Film Bureau, $165.

Beginning with a display of symbols like "stop" and "go" lights, highway markers, and other signs which give directions, this film goes on to identify symbols for things, names of people and places. The development of some symbols is reviewed, such as word and letter symbols. In spite of this introduction, the use of symbols in the remainder of the film is very confusing particularly with respect to constants, variables and numerals.

The language used in this film is mathematically poor. No mention is made of basic algebraic principles which results in statements like "2 is removed from the parenthesis." The reviewers feel that the film is not usable in algebra courses.

LANGUAGE OF ALGEBRA. See Junior High Film Series.

LANGUAGE OF GRAPHS. 1948. sd, bw, 13 min., jh; Coronet Instructional Films, $62.50, co $150.

Practical uses of bar graphs, line graphs, circle graphs, and graphs of a function are illustrated. The value of a graph in telling a story with pictures is shown as well as the algebraic relations between x and y values on a straight line. Insight into some business principles is also offered.

This film will be of particular use to students studying some simple applications of mathematics.

THE LANGUAGE OF MATHEMATICS. 1950. sd, bw, 11 min., jh; Coronet Instructional Films, $60, co $120.

A fire drill is used as a starting point in a simple discussion of the use of precise information in mathematics to solve problems. The language of mathematics is shown to be a way of communicating ideas as in blueprints, graphs, and other forms.

The issues considered in the film suggest that a more appropriate title would have mentioned something about precise measurements.

LARGE ANGLES AND COORDINATE AXES. See Trigonometry Series.

LAW OF COSINES. See Trigonometry Series.

LAW OF SINES. See Trigonometry Series.

LAW OF TANGENTS. See Trigonometry Series.

LINEAR EQUATIONS IN ONE UNKNOWN. See Advanced Algebra Series.

LOGARITHMIC OPERATIONS, I, II. See Engineering Problems.

LOGARITHMIC SYSTEMS. See Engineering Problems.

LOGARITHMS AND THE SLIDE RULE. 8 films, 1961. sd, bw; sh, jc, a; International Film Bureau, Inc., $795.

This series of eight films might be usable in technical courses in which skills alone are desired. Certainly no one should use this series who wants to derive mathematical understanding of logarithms and the slide rule. The four films on slide rule are more satisfying than the four on logarithms probably because we view the slide rule as more of an instrument than a mathematical idea. The most serious criticism of the series is the disregard for standard notation and the reliance on rote learning of rules that are often very arbitrarily conceived.

Logarithms and the Slide Rule—Lesson I. 30 min., $125; sh, jc, a.

The series opens with a general discussion of the need for speed in computation. The slide rule and digital computer are mentioned as important advances in the area of computation, and a typical slide rule is displayed as well as several other types including the spiral type with a 500-foot scale.

After a problem in regular multiplication, the rules of exponents are reviewed. The possibility of expressing numbers as powers of ten and multiplying by adding exponents is shown by example. It is not mentioned that 16.53 is only approximately equal to $10^{1.21827}$. A definition of logarithm is given but the base b is not restricted as it needs to be for this discussion. For example, it should be pointed out that one is not a logical choice for b. Using the rules for exponents, the parallel rules for logarithms are presented. The traditional procedure for determining the characteristic by noting the powers of ten which bracket the given numeral is given but the rule "one less than the number of digits to the left of the decimal point" is entirely inadequate. A number "partly positive and partly negative" is invoked to explain the method for determining the characteristic of numbers less than one. Furthermore, it is not mentioned that these numbers are restricted to the non-negative numbers.

Logarithms and the Slide Rule—Lesson II. 30 min., $125; sh, jc, a.

Following an extensive review of Lesson I, several examples are worked to practice using the rule for numbers greater than one. The reason given that numbers 68.0 and 680 have the same mantissa is that these numbers are "the same distance along the way" between 10 and 100, 100 and 1000, respectively.

The method of adding and subtracting 10 to a logarithm for convenience of operation is presented. Phrases such as "0.0068 is penned in between 0.01 and 0.001" and "a number farther in the hole" are used in the development which indicate the lack of precision in language and the general intuitive aspect of the series.

Logarithms and Slide Rule—Lesson III. 30 min., $125; sh, jc, a.

An extensive review of the first two lessons is followed by a simplification of

$$\sqrt[3.2]{\frac{(6.278)\ (2.06)}{(92.563)\ (0.00806)}}$$

using a logarithmic scheme to ease computation. No justification is given in many cases with complete dependence on rules.

An elementary method for devising entries in the logarithm table is presented. It is shown that logarithms are in arithmetic progression while the corresponding powers of ten are in geometric progression. The general development here is good although until this point the instructor has not mentioned that he deals with approximations.

Logarithms and the Slide Rule—Lesson IV. 30 min., $125; sh, jc, a.

A lengthy review of the method for computing any power of ten could be considerably shortened by noting that, for example, $10^{2.5} = \sqrt[2]{10^5}$.

The extraction of the square root of 100,000 by means of the algorism is done in detail but does not seem to be of interest here and could be deleted.

A logarithm table is examined carefully and the entries explained. An example is shown using the table but the definition of significant digits is inadequate. An example of the method of finding an anti-logarithm is carried out, but the rule for interpolation is simply given without any justification. In fact, interpolation is treated much too briefly and the student who knows little about the technique will find the film of no help.

Logarithms and the Slide Rule—Lesson V. 30 min., $125; sh, jc, a.

The discussion of the slide rule begins in this film. The C and D scales are identified and the relative positions of the numerals on these scales are explained in terms of logarithms. In this explanation the lecturer says "the little 9 is less than the large 2" which leaves something to be desired. Several examples of multiplication are shown including some in which the problem of "going off the scale" appears.

Logarithms and the Slide Rule—Lesson VI. 30 min., $125; sh, jc, a.

After a review of the previous film, the C and D scales are used in division and the "off the scale" problem is handled for division. The principle of proportion in the use of the slide rule is discussed in some detail and several problems involving proportions are solved. Examples of problems in which both multiplication and division are involved are carried out with good technique. The rules for multiple settings on the slide rule are discussed.

Logarithms and the Slide Rule—Lesson VII. 30 min., $125; sh, jc, a.

The CI scale is discussed and several problems involving this scale are solved. The use of the slide rule in reciprocals, multiplication, and division is demonstrated. The CI scale is emphasized because it shortens calculation time by reducing the number of settings required.

The A and B scales are explained and used for squaring and extracting square roots.

Logarithms and the Slide Rule—Lesson VIII. 30 min., $125; sh, jc, a.

The A scale, settings, square root, and result reading are reviewed. The rule for locating the decimal point is given but could be handled more easily through the use of scientific notation. The decimal point is "moved around to the left" indicating poor use of language. The K scale and cube roots are discussed but the method for locating the decimal point is not good.

LOGARITHMS—CHARACTERISTICS. See Engineering Problems.

MCGRAW-HILL TEACHER EDUCATION SERIES. 5 films, 1959. sd, bw; jh, sh; tchrs. of el, jh, sh; guide; McGraw-Hill Text Films, $600.

Sentences and Solution Sets. 21 min., $140; jh, sh; tchrs. of el, jh, sh; guide.

The instructor points out that the concept of set plays a very basic role in our daily lives. He claims that the major advantages accruing from the use of sets in mathematics are clarification, simplification, and unification. By means of a somewhat artificial classroom scene, the idea of an open sentence is presented. The terms "set selector," "subset," "universal set," and "solution set" are illustrated. A variable is defined precisely and this definition is contrasted with the various vague descriptions given in many traditional algebra classes.

We recommend this film very highly for viewing by all teachers and prospective teachers of elementary and secondary mathematics.

Concept of a Function. 16 min., $105; jh, sh, jc, sc; tchrs. of jh, sh; guide.

In this film the lecturer develops the notion of function from the set concept. Sentences in two variables are discussed including the graphs and solution sets of such sentences. This background serves as preparation for the definition of function which is defined as a set of ordered pairs such that no first element can appear with different second elements. Domain, range, and rule of a function are defined with emphasis on the point that a function is not a formula but rather a set of ordered pairs. The expression

$F(x)$ is defined as the value of the function at x or the value of y. We feel it would be more consistent with the definition of function to define it as the second element of the ordered pair whose first element is x.

The lecturer succeeds in presenting sound mathematics and uses his time efficiently. Although the films are designed for teacher training we feel this film could be used with high school students who have some background in contemporary mathematics.

Irrational Numbers. 23 min., \$150; jh, sh, jc, sc; tchrs. of jh, sh; guide.

"An irrational number is the square root of a number that we can't take the square root of." The lecturer submits that a statement such as this would more often than not summarize an algebra student's knowledge of irrational numbers. He then demonstrates, with the help of classroom scenes, that the road to far greater understanding is not a rocky one and is certainly worth traveling.

The film emphasizes the importance of the study of the decimal expansions of numbers. A rational number is defined as a number which is the quotient of two integers. Also discussed are properties of order and denseness in the set of real numbers.

Except for a puzzling statement about the period of a repeating decimal, namely: "We say the decimal has period 2 because it starts to repeat after the second place," this film is well done. It should prove quite valuable if put to use in preservice and in-service teacher training.

Number Fields. 17 min., \$115; sh, jc, sc; tchrs. of jh, sh; guide.

The film opens with a dialogue between a student and his teacher concerning rationalizing the denominator of $(12-7\sqrt{3})/(3-2\sqrt{3})$. The scene then shifts to the regular lecturer who indicates that the teacher would have been better prepared to give answers to such questions as "Why do we rationalize the denominator? Will we always get an answer of the form $a+b\sqrt{3}$?" if she had been well versed in the concept of number fields.

Then follows a quick review of the concept of set and properties of an operation. A definition of closure is carefully given. A field is defined and the laws concerning its operations are listed.

The set of integers is used as an example of a set which is not a field. The rational numbers, real numbers, and complex numbers are shown to be fields before returning to the numbers whose numerals are of the form $a+b\sqrt{3}$ and this set is shown to be a field also.

The presentation is well done but we feel that number fields is too large a topic for a film of this length. We also feel that better motivation could have been provided for the topic although this does not detract from its effectiveness. The film can be used for in-service education or teacher training if the instructor is careful to give further examples including finite fields.

Patterns in Mathematics. 14 min., \$90; jh, sh; tchrs. of jh, sh; guide.

The opening scene of the film shows the instructor in a discussion of the essential nature of mathematical problems. He notes that while the mathematician looks for patterns, the high school student often looks at mathematics as a bag of tricks. High school mathematics should be a study of patterns, says the instructor, since this is the essence of the "new mathematics." Some of the patterns discussed are the distributive law, commutative law for multiplication, graphs of straight lines, and the solving of simultaneous linear equations. The point is made that patterns develop ability to generalize, understand relationships, improve insight, power, and readiness for further study.

This would be an excellent film for teacher-education classes or seminars. The reviewers are especially pleased by the accuracy of the mathematics in this film and series.

MANTISSAS. See Engineering Problems.

THE MEANING OF PI. 1949. sd, bw, 12 min., jh, sh; Coronet Instructional Films, \$60, co. \$120.

The terminology for circles is introduced and illustrated with several objects. It is shown that it takes a "little more" than three diameters to give the circumference of a circle and it is seen that as diameter increases, the circumference increases. Further comparison shows that the ratio of circumference to diameter is slightly more than 3.14. A brief history of several notations for expressing π is given. It is emphasized that the value $3\frac{1}{7}$ is chosen for convenience only.

MORE SOLUTIONS OF LINEAR EQUATIONS. See Intermediate Algebra Series.

MULTIPLICATION AND DIVISION. See Engineering Computation Skills: The Slide Rule.

MULTIPLICATION OF RATIONAL NUMBERS. See Intermediate Algebra Series.

MYSTERIOUS X. See Adventures in Number and Space.

NATURE OF LOGARITHMS. See Advanced Algebra Series.

NEW NUMBERS. See Understanding Numbers.

NUMBER FIELDS. See McGraw-Hill Teacher Education Series.

THE NUMBER SYSTEM AND ITS STRUCTURE. 1961. sd, co, bw, 11 min.; jh, sh; Coronet Instructional Films, bw, \$60, co. \$110, guide.

After a brief history of number which includes the notion of place holder, concepts concerning the number system are discussed. The property of closure, the commutative and associative laws, and the distributive law are included in this discussion.

For a review of our number system's structure and fundamental principles, this film should be excellent. However, if used as an introduction, too much will be covered to allow good understanding. There are some errors which will need correcting by the teacher.

NUMERATION SYSTEMS. See Junior High Film Series.

OBLIQUE TRIANGLES. See Engineering Computation Skills: The Slide Rule.

PATTERNS IN MATHEMATICS. See McGraw-Hill Teacher Education Series.

PERMUTATIONS AND COMBINATIONS. See Advanced Algebra Series.

PRACTICAL USE OF LOGARITHMS. See Trigonometry Series.

PROGRESSIONS, SEQUENCES, AND SERIES. See Intermediate Algebra Series.

PROPORTIONS AT WORK. 1961. ad, co, bw, 12 min., jh, sh; International Film Bureau, Inc. $120.

A biologist at work using proportions serves as motivation for this film. A vague definition of ratio is given and some properties of proportions are dealt with briefly. The rule that the product of means equals product of extremes is derived but done too quickly for most students to follow easily. Units of linear and area measure are included but their use in proportions follows the pattern used in many physics courses.

The film does not seem to cover too much material and, if used, could promote considerable classroom discussion in general mathematics or algebra classes.

PYTHAGOREAN THEOREM: THE COSINE FORMULA. 1960. sd, bw, 5½ min., sh; Coronet Instructional Films, $30.

This film uses animation to illustrate the derivation of the law of cosines. The Pythagorean theorem is then derived as a special case of this law by algebraic methods and by geometric animation.

The pace of the film may be too fast for average students and the development may be too rigorous for most students to follow if used as an introduction.

PYTHAGOREAN THEOREM: PROOF BY AREA. 1960. sd, bw, 5½ min.; jh, sh; Coronet Instructional Films, $30.

The Pythagorean theorem is stated and a special case of an isosceles right triangle is considered. The equality of the areas of parallelograms with equal bases and altitudes is considered and the notion used to justify the Pythagorean theorem for any right triangle. Examples are illustrated.

The animated demonstrations given are good but of course, do not constitute proofs. The pace of the film is probably too fast for the average eighth grader who sees these ideas for the first time but might be adequate for reviewing.

QUADRATIC EQUATIONS. See Intermediate Algebra Series.

QUICKER THAN YOU THINK. See Adventures in Number and Space.

RADICALS AND THE REAL NUMBER SYSTEM. See Intermediate Algebra Series.

RAISING NUMBERS TO POWERS. See Engineering Computation Skills: The Slide Rule.

RATIO PROBLEMS. See Engineering Computation Skills: The Slide Rule.

THE RATIONAL NUMBERS. See Junior High Film Series.

RIGHT TRIANGLE APPLICATIONS. See Engineering Computation Skills: The Slide Rule.

RIGHT TRIANGLES. See Engineering Computation Skills: The Slide Rule.

RIGHT TRIANGLES AND TRIGONOMETRIC RATIO. See Trigonometry Series.

ROOTS AND EXPONENTIAL EQUATIONS. See Engineering Computation Skills: The Slide Rule.

ROOTS OF HIGHER ORDER. See Intermediate Algebra Series.

SENTENCES AND SOLUTION SETS. See McGraw-Hill Teacher Education Series.

SHORT CUTS. See Understanding Numbers.

SIGN LANGUAGE. See Adventures in Number and Space.

SIMILAR TRIANGLES IN USE. 1961. sd, co., 11 min., jh, sh; International Film Bureau, Inc., $120.

Two examples of the use of similar triangles and their proportional sides are given to illustrate practical applications. Trigonometry is coupled with similar triangles and some of the special tools of occupations using these ideas are shown.

It is probably the case that most teachers could stage situations that would be at least as effective as those in the film. The emphasis on trigonometry as a tool of the occupations illustrated is probably misplaced in the light of present use of trigonometric functions.

Simplifying Complex Fractions. See Advanced Algebra Series.

Slide Rule. See Special Lessons in Physics.

Solids in the World Around Us. See Discovering Solids.

Solution of Equations Beyond the Second Degree. See advanced Algebra Series.

Solving Equations in Fractional Form. See Intermediate Algebra Series.

Solving Problems With the Quadratic Formula. See Advanced Algebra Series.

Solving Simultaneous Linear Equations. See Intermediate Algebra Series.

Special Lessons in Physics. 3 films, 1957. sd, bw, sh; tchrs. of sh; Encyclopedia Britannica Films, Inc., bw, $495 co $990; guide.

Elements of Trigonometry. 30 min., sh; tchrs. of sh; bw $165, co. $330.

Ideas on significant figures are discussed at the outset, but unless the viewer is already familiar with these notions, we feel he will have some difficulty in following the discussion. A large demonstration slide rule is effectively used to examine the parts and scales of the slide rule. Multiplication and division are carefully discussed as well as the computation of squares and square roots.

The closeups of the slide rule are better than would be possible in the usual classroom demonstration. No attempt is given to show the rationale behind the processes. We feel that the film might be useful as an introduction to the slide rule but that the pace of presentation is too fast.

Algebra and Powers of Ten. 30 min., sh, tchrs. of sh; bw $165, co. $330.

Equations as pictures of experiments are considered. A balance is used to demonstrate some transformations of special equations. A term of the type $\frac{1}{a}+\frac{1}{b}$ is simplified. Some common errors that occur in the treatment of literal equations are pointed out. A model is shown in an attempt to illustrate the square of a binomial. A definition of exponent and several examples with base ten are discussed. Scientific notation is discussed as a practical way of writing very great or very little numbers.

The use of a balance in transforming equations is not very good. The treatment of fractional equations involves much symbol juggling. The model used to illustrate the square of a binomial needs dark and light shading of its parts since it is difficult to see. Scientific notation is very hastily introduced and, although examples are cited, no definition of negative exponents is given. The definition given for exponent is very poor. We do not recommend that this film be used in secondary schools.

Slide Rule. 30 min., sh; tchrs. of sh; bw $165, co. $330.

An angle is defined and a specific angle is constructed. Using the right triangle, definitions of sine, cosine and tangent are given. If a and b are the acute angles of a right triangle then it is seen that $a+b=90°$ and $\sin a=\cos b$. An experiment is performed to examine the ratio of sides to fixed hypotenuse of 50 centimeters. Projections of line segments on a given line are discussed.

The visual aids used in the film are very good. The material presented will be incomplete for those who have never studied trigonometry and is, in general, hastily presented. After the experiment described above, no mention is made of the purpose of a table of such ratios. The film would do a fair job of reviewing sine, cosine, and tangent.

Special Products and Factoring. See Intermediate Algebra Series.

Squares, Cubes, and Roots. See Engineering Computation Skills: The Slide Rule.

Standard Techniques of Factoring. See Advanced Algebra Series.

Stretching Imagination. See Adventures in Number and Space.

Surface Areas of Solids I and II. See Discovering Solids.

Symbols in Algebra. 1961. sd, bw; jh, sh: Coronet Instructional Films, bw $60, co. $120 11 min.; guide.

An introduction shows that the students have been using formulas in arithmetic and that the basic purpose of algebra is the establishing of general rules such as these formulas. The film shows how letter symbols are used in much the same way as numerals in arithmetic and concludes with an example showing the use of a letter symbol as an unknown in an equation.

The best use of this film would be to introduce a unit on algebra in Grade 8 since it does a good job of relating algebra to arithmetic. The solving of equations is handled only briefly.

Tables of Trigonometric Ratios. See Trigonometry Series.

Theory of Equations and Synthetic Division. See Advanced Algebra Series.

Time. 1959. sd, co, bw, 15 min.; el, jh, sh, jc, a; tchrs. of el, jh, sh; Indiana University, bw $75, co. $150.

The use of time in daily living is the theme of this film. The sun is shown to be one of man's oldest time pieces. A detailed treatment is given to the development of the time zones with an animated sequence of a rocket circling the earth being used to illustrate the necessity for the International Date Line. Daylight Saving Time is explained. A discussion follows on how astronomers take photographs of stars' paths to determine time and the film concludes with a summary of the ideas discussed.

The animation and models are excellent. The inclusion of an examination of pendulums is well done and the familiar examples given are good. The use of a rocket to show the need for the International Date Line makes this idea easy to understand. This is one of the best films this committee has seen.

TRIGONOMETRIC APPLICATIONS. See Engineering Problems.

TRIGONOMETRIC RATIOS AS PERIODIC FUNCTIONS. See Trigonometry Series.

TRIGONOMETRIC SCALES. See Engineering Computation Skills: The Slide Rule.

TRIGONOMETRY AND SHADOWS. See Trigonometry Series.

TRIGONOMETRY OF LARGE ANGLES. See Trigonometry Series.

TRIGONOMETRY MEASURES THE EARTH. See Trigonometry Series.

TRIGONOMETRY SERIES. 21 films, 1960. sd, bw, sh, jc; Modern Learning Aids, $3,150.

This series may be useful in technical, vocational, or service courses where the theory underlying trigonometric skills is unimportant. There are a few exceptions to this statement as will be noted in the individual film reviews, but it is generally the case that skill in techniques receives greatest emphasis. Therefore, the series is not to be recommended for contemporary curriculum programs.

Pedagogically, we feel that too much information is given to be memorized. Certainly some facts need to be memorized but the quantity required here is excessive. There are inconsistencies in notation, for example, 'N.D.' is used for the words 'not defined' in an early film and the symbol for infinity is used later. The emphasis on what not to do is questionable and yet stress is often placed on procedures that are not to be done. Repetition is also useful in teaching when used sparingly but often items are repeated to the limit of boredom. It seems to this committee that the most important constructions should be made in the presence of the students and yet in these films it is often the case that these constructions are already on the chalk-board at the beginning of the film. This leaves the viewer with no idea as to the development of the notion. This committee objects to the colloquial language used in the series. Such terms as "shuffling fractions," "divided out," "penned in," "knock out," "cleaning up," "opening parenthesis," "this number is really real," "the sine of zero is nothing at all," "a segment is capable of," "the square roots will lift off," and many more, are not acceptable in a series designed for wide distribution.

Definitions are, in general, poorly stated, if indeed given at all. The logical structure of trigonometry is not well presented in spite of the stated intent of making trigonometry an extension of geometry. No clear distinction is made between convention and definition. As seen above, the language used is imprecise and the mathematics suffers as a consequence.

Trigonometry and Shadows. 26 min., $150; sh, jc, a.

A review of various applications of trigonometry—surveying, construction, navigation, warfare, and cartography—is given briefly around the early history of trigonometry. The significance of the ratio of the lengths of the sides of a triangle is stressed, and the sine of an angle is defined.

The historical comments at the beginning of the film are good although most teachers would be able to provide similar comments of their own. The drawing illustrating Thales' solution to measure the inaccessible height of a pyramid seems to be too intricate. The notion of a standard triangle comes very fast and it will be difficult for most students to follow. The ideas of sine as a ratio is not the notion which is currently being used. This is also true of the other functions which are discussed here.

Right Triangles and Trigonometric Ratio. 29 min., $150; sh, jc, a.

Thales' method for finding the height of a pyramid using the ratio of corresponding sides of similar triangles is generalized in the definitions of the sine, cosine and tangent of an angle as ratios of the sides of a triangle having one right angle. The values of these functions for angles of 30, 45, and 60 degrees are derived.

This is an extension of the traditional treatment found in the first film. The statement that the square root of 3 is 1.732 implies that the square root of 3 is a rational number. Other errors are noted such as a line segment being called a ratio. Defining "function" as "depends on" is extremely confusing. This presentation of trigonometry will hardly be in line with any of the contemporary programs.

Using Sines, Cosines and Tangents. 29 min., $150; sh, jc, a.

After a brief review of the definitions and values of the sine, cosine, and tangent functions for 30, 45, and 60 degree angles, a series of examples is demonstrated. Emphasis is placed on proper choice of function to make the solution easiest. An arc of a unit circle is defined and used to expand the relationships among the simple trigonometric functions and to show that these functions are linear.

The instructor wisely guides the observer to consider the plausibility of answers that are obtained and uses some good illustrations. The unit circle is used in the classic manner to define the functions rather than to use the unit circle coordinates and a winding function. The nonlinearity of the tangent function is not made clear. The accuracy of the statements about the history of the word "sine" is doubtful.

Trigonometry Measures the Earth. 28 min., $150; sh, jc, a.

The film illustrates the power of trigonometry in the solution of complex and difficult problems. It shows in some detail how Eratosthenes measured the circumference and diameter of the earth. A method of finding the distance to the moon is also described.

The reaction of the reviewers to this film is favorable and they recommend its use in any of the different types of trigonometry courses—contemporary or traditional. It provides excellent historical interpretation of the applications of trigonometry. It is suggested, however, that the teacher using this film carefully preview it and make reference to the fact that the lecturer does not refer to the diagrams as triangles. The use of props is definitely primitive.

Cosecant, Secant and Cotangent. 27 min., $150; sh, jc, a.

The cosecant, secant, and cotangent functions are defined and shown to be the reciprocals of the sine, cosine, and tangent functions. The representation of these functions as line lengths associated with the unit circle is described in some detail and the origin of the names of the various functions is given. Values of these functions are developed for angles of 0, 15, 30, 45, 60, 75, and 90 degrees. The complete range of trigonometric tables is hinted at but not yet developed.

Inconsistency in language is persistent. The terms "trigonometric functions," "trigonometric relationships," "trigonometric ratios," and even "trigonometric segments" are used synonymously. The observer may derive the impression, from the phraseology of the instructor, that sine is an increasing function and cosine is a decreasing function.

Eight Fundamental Trigonometric Identities. 28 min., $150; sh, jc, a.

This film was not reviewed. However, the publisher's description is given here.

The Pythagorean relation for right triangles is used to develop three trigonometric identities involving the squares of the simple trigonometric functions. The two ratio relationships for tangent and cotangent in terms of sine and cosine, and then the three reciprocal relationships defined in the previous film complete the eight fundamental trigonometric identities. Several examples of the use of these identities for simplifying complex trigonometric relationships are worked out in step-by-step detail.

Working With Trigonometric Identities. 29 min., $150; sh, jc, a.

A brief review of the eight fundamental identities serves as a basis for analysis of more complicated identities. Techniques such as working on only one side of the identity, watching the form of the terms, and checking results are shown in detail. Geometric illustrations of the identities are developed on the unit circle.

The instructor does not make it clear that identities are theorems and fails to write these proofs in such a way that the form of the proof is clear. The fact that the domains of the eight fundamental identities have not been restricted allows for some errors in conclusions. There are also occasional errors on the chalkboard both in mathematics and in pedagogical efficiency. The terminology is often colloquial and idiomatic and, in too many cases, the language is not precise.

Tables of Trigonometric Ratios. 29 min., $150; sh, jc, a.

Conventional tables of trigonometric functions are described and examples of their use illustrated. This film could be used in some courses to supplement a lecture on the use of trigonometric tables.

The use of materials in this film is good. On the other hand, the pedagogy is questionable, particularly when stress is placed on things that should not be done. Moreover, great emphasis is placed on memorization of large amounts of material.

Interpolation in Trigonometric Tables. 28 min., $150; sh, jc, a.

After a review of the form and organization of trigonometric tables, the technique of reading such tables is described with care. Examples are worked out in detail, showing how to carry out interpolation and inverse interpolation procedures. The importance of careful organization of one's work, and the errors to be watched for and avoided, are considered.

This film is a good summary of the use of trigonometric tables. The methodology in the film indicates that the development is reflecting a cookbook version of trigonometry. This is particularly shown by the rigorous patterns that are required for interpolation.

Introduction to Logarithms. 28 min., $150; sh, jc, a.

Logarithms are introduced as a useful mathematical technique for carrying out the computational manipulations required in solving complex trigonometry problems. A logarithm is first defined as an exponent in the general sense that if $N=B^P$, then $\log_B N=P$, where P is the logarithm of N to the base B. Examples of the use of logarithms are then worked out, and in the process the definitions of mantissa and characteristic are developed.

The first rule for determining the characteristic involves "counting the number of digits to the left of the decimal point, and subtracting one."

Practical Use of Logarithms. 30 min., $150; sh, jc, a.

Various applications of the use of logarithms are presented including problems of multiplication, division, raising to powers, and extracting roots. All discussion is restricted to base ten and the emphasis is on the service aspect of trigonometry rather than as an important branch of mathematics.

Using Logarithm Tables. 29 min., $150; sh, jc, a.

Finding the mantissa of a logarithm from a three-place table and then the reverse process of finding a number given its logarithm, is shown. A brief discussion of significant digits is applied to logarithms and antilogs. Examples are worked out showing how to interpolate in log tables, and how to do inverse interpolation. The log of a trigonometric value is found in a table, and the existence of log trig tables is presented briefly. Finally, the nature of the scales and the use of a slide rule are discussed.

The major part of the film is devoted to the continuation of the study of logarithms that was begun earlier. The section on the nature and use of the slide rule will provide the user with no help in teaching the slide rule.

Large Angles and Coordinate Axes. 30 min., $150; sh, jc, a.

Many applications of geometry and trigonometry involve angles greater than 90°. This film defines and describes these large angles and their trigonometric values. Positive and negative angles, the four quadrants of a plane, and the coordinates of a point in any of the four quadrants are discussed. The values of the sine, cosine, and tangent (with proper signs) are defined for all quadrants. Methods of reducing the large angles to equivalent smaller ones for the use of trigonometry tables are explained.

Trigonometry of Large Angles. 30 min., $150; sh, jc, a.

After a review of the nature of large angles, it is shown that all the definitions of the trigonometric functions in terms of R, x, and y apply to any large angle if proper care is taken to identify the sign of the function. Thus, trigonometric functions are signed quantities. All eight of the simple trigonometric identities are shown to work in all four quadrants. Finally, a number of examples are worked out in detail showing the reduction of functions of large angles to small, and of functions of negative angles to equivalent functions of positive angles.

Again, there is a tendency to ask the student to memorize too much extraneous material, such as the "CAST" rule for remembering the positive functions in each of the four quadrants.

Law of Sines. 30 min., $150; sh, jc, a.

The equation for the Law of Sines is derived and then used to solve a problem in which two angles and the included side are known. The solution provides directly the values for all three sides and all three angles. Finally, an ambiguous problem is described in which are given two sides and the angle opposite one of them. Either of two solutions is possible. A single solution would occur if one unknown angle is a right angle.

The reviewers feel that this film could possibly be used separately from the sequence. However, there is a continuing use of sloppy language. The derivation of the Law of Sines may be a bit sophisticated. It is geometric in nature, however, and in this respect, ties in well with background materials in the other films of the series.

Law of Cosines. 30 min., $150; sh, jc, a.

A derivation of the Law of Cosines is shown and its application to a problem is carefully worked out. The problem of three given sides is worked out. The tedious calculation is pointed out and the advantages of logarithms in such situations are made obvious. However, the Law of Cosines is not written in a form suitable for the use of logarithms.

This film might be used as a supplement to the "in-class" teaching of the Law of Cosines. The film is not as good as the one on the Law of Sines. Extreme monotony arises in this film because of the large amount of repetition. The film would be more useful if it had been more carefully edited.

Law of Tangents. 28 min., $150; sh, jc, a.

The Law of Tangents is presented and used in a practice problem. The derivation of the Law of Tangents is quickly done with the help of previously prepared figures and equations.

Law of Tangents could be used as a supplementary film to a lecture on the Law of Tan gents or as a filler on that particular topic in case the teacher chooses this method. It is a good film for enrichment for able students as it presents an unusual geometric proof of the Law of Tangents.

Trigonometric Ratios as Periodic Functions. 28 min., $150; sh, jc, a.

Conic sections, periodic and harmonic motion, and the general form of the sine curve are discussed. The general presentations seem to be satisfactory although since this is the first consideration of the sine function as a periodic function, the development may be difficult for the average student to follow. This is one of the unfortunate consequences of this kind of development. Several definitions are not clearly given and such terms as "period of a function" and "amplitude" are only incidentally mentioned. The graph of $y = A \text{ Sin } Bx$ is not carefully explained. The film may be useful in some situations but the pace will be too fast for most students.

Graphs of Periodic Functions. 29 min., $150; sh, jc, a.

Further discussion on $y=A \sin Bx$ is presented as well as radian measure. Plotting periodic functions in terms of radians is described, and the sine curve is contrasted with the tangent curve.

The use of this film is not recommended in any class because of the extremely poor terminology that is used in it. To cite an example, the amplitude of a sine curve is defined as "the thickness and thinness of the up and down." The relation $s=x \cdot r$ (where s represents the arc length, x the number of radians in the central angle, and r the radius) is given without any appeal to intuition or proof. The student is asked to memorize that "π is equal to 180°" and later the statement is given that π radians is equal to 180°.

Addition Formulas and DeMoivre's Theorem. 28 min., $150; sh, jc, a.

This film is a rapid survey of several applications of trigonometry to algebraic equations. Complex roots of a cubic equation are derived from DeMoivre's Theorem which is itself derived from addition formulas for trigonometric functions.

The film covers entirely too much material to be conveniently understood by most students. DeMoivre's Theorem is not well established and the formula for the sine of the sum of two angles is developed for first quadrant angles only.

Double and Half Angle Formulas. 28 min., $150; sh, jc, a.

The addition formula developed in the previous film is used to derive the double-angle formulas for sine, cosine, and tangent. The use of these formulas in simplifying identities is worked out in a problem. The half-angle formulas for sine, cosine, and tangent are derived and used to simplify another identity.

This film presents nothing that could not be presented by a regular classroom teacher. The developments are standard for the double and half-angle formulas. In the lecturer's summary, he mentions that he has tried to illustrate the logical and psychological unity of trigonometry. The logical unity was certainly not apparent; neither was the psychological!

UNDERSTANDING NUMBERS. 7 films. sd, bw; jh, sh, jc, sc, a: tchrs. of el, jh, sh; University of Michigan TV, $700.

The Earliest Numbers. 30 min., $100; jh, sh, jc, sc, a; tchrs. of el, jh, sh.

Some early numeration systems are discussed with emphasis on the Egyptian and Babylonian systems. A good demonstration of the use of a counting board in computation is given. The relationship between number and language is considered.

The historical development is well done, although at one point the Egyptian and Babylonian systems were interchanged and at a later point in the film Arabic symbols were used in place of Egyptian symbols. No differentiation was made between symbols used to compute and symbols used to record in the systems discussed. This committee feels that the film definitely covers too much historical ground and that the summary, which is given orally by the instructor, fails to tie up the ideas given. Overall, the film is well done in spite of these weaknesses and it would certainly be worthwhile as a film on the history of numeration systems.

Base and Place. 30 min., $100; sh, jc, sc, a: tchrs. of el., jh, sh.

A presentation of the binary system and a demonstration of the use of the system in the digital computer are given in this film. The most outstanding feature of the film is the fine treatment of the binary system and its relationship to the development of digital computers. The lecturer fails to mention the relationship between the base and the number of distinct digits.

Big Numbers. 30 min., $100; jh, sh, jc, sc, a; tchrs. of el, jh, sh.

The film illustrates and demonstrates the use of scientific notation. A rather extensive discussion of perfect numbers is also included.

The title of the film is slightly misleading since the discussion also centers on lesser as well as greater numbers. As a teaching instrument, the film will probably not serve too well, although it might be used in a mathematics club or for any occasion of general interest. The statement that only fifteen perfect numbers are known is no longer correct and will clearly date the film.

Fundamental Operations. 30 min., $100; jh, sh, jc, sc, a; tchrs. of el, jh, sh.

The fundamental operations of addition, multiplication, subtraction, and division are the main issues here as well as the postulates associated with each of these operations. Subtraction is shown to be the inverse of the operation addition and similarly for multiplication and division. Addition and multiplication tables are given for modulo 5 and used to carry out a few exercises. Discussion is briefly extended to the rational numeration systems.

We recommend the film as a review of a section on modular arithmetic or as a preview of such a section. The emphasis on the patterns and fundamental operations in mathematics and the use of modular arithmetic to demonstrate these is good. The analogy of a "commutor" and the commutative law did not seem satisfactory to this committee. The tables for addition and multiplication in modulo 5 are introduced without sufficient development leaving the student to wonder where they came from.

Short Cuts. 30 min., $100; sh, jc, sc, a; tchrs. of jh, sh.

This film explains and illustrates some mathematical short cuts as a means of simplifying computation. The grating or gelosia method of multiplication, the principles of logarithms, and the slide rule are considered.

We suggest that this film be used for review rather than as an introduction to the topics given here. Use is made of the notion that logarithms are exponents prior to the statement of this relationship. The instructor mentions that to multiply numbers it is necessary to add exponents, but neglects to restrict this operation to numbers with a common base. No summary is given and the over-all organization of the material is poor.

Fractions. 30 min., $100; jh, sh, a; tchrs. of el, jh, sh.

The fable concerning the distribution of nineteen head of cattle by the portions $\frac{1}{2}$, $\frac{1}{4}$, and $\frac{1}{5}$ is used to introduce fractions. A common fraction is defined and its five different meanings are analyzed with visual aids. The history of fractions is reviewed from the viewpoint of number theory. A chart is used to illustrate several types of fractions such as decimals, rational fractions, and duodecimal fractions. Some efforts are made to elucidate the meaning of the fractions and the relations between them. Attention is also given to repeating decimals.

The exploration of the five implied meanings of a fraction is adequate and stimulating with the use of visual aids being very helpful. The brief review of the history of fractions is satisfactory. The film seems appropriate for a course in mathematical appreciation for a group with widely different preparation in mathematics. The level of material is uneven and the range of difficulty is wide, although the presentation of the topics is not rigorous. Little attention is given to operations with, and applications of, fractions. The lecturer makes occasional slips in writing and in talking which indicate a need for more editing.

New Numbers. 30 min., $100; sh, jc, sc, a; tchrs. of sh.

This film introduces the student to some of the new numeration systems. Rational, irrational, complex, and transfinite numbers are discussed in some detail. A good demonstration of the meaning of a one-to-one correspondence is given. The film provides the student with an excellent opportunity to consider numbers other than the familiar real numbers. A great deal of emphasis is placed on the proper naming of numbers.

The lecturer occasionally uses the word "number" when he should say "numeral." Many of the mathematical terms used, "cardinal" for example, are casually mentioned without adequate development or further use. The oral presentation is too rapid and the chalkboard work is not up to the par established in the other films of the series.

USING FRACTIONAL AND RATIONAL EXPONENTS. See Intermediate Algebra Series.

USING LOGARITHMS IN PROBLEMS. See Advanced Algebra Series.

USING LOGARITHM TABLES. See Trigonometry Series.

USING LOGARITHMS TO SOLVE EQUATIONS. See Intermediate Algebra Series.

USING SINES, COSINES AND TANGENTS. See Trigonometry Series.

VARIATION: A LESSON IN READING. See Intermediate Algebra Series.

VOLUME AND ITS MEASUREMENT. 1960. sd, bw, 11 min.; el, jh; guide, Coronet Instructional Films.

Beginning with definitions of volume, formulas for volumes of rectangular solids, prisms, and pyramids are developed using plastic models. The need for a unit of measurement is stressed. Although the development of the volume formulas for rectangular solids and triangular prisms was incomplete, the film might be useful in junior high school classes on intuitive geometry.

VOLUMES OF CUBES, PRISMS, AND CYLINDERS. See Discovering Solids.

VOLUMES OF PYRAMIDS, CONES, AND CYLINDERS. See Discovering Solids.

WHAT'S THE ANGLE? See Adventures in Number and Space.

THE WHOLE NUMBERS. See Junior High Film Series.

WORKING WITH POSITIVE AND NEGATIVE EXPONENTS. See Intermediate Algebra Series.

WORKING WITH TRIGONOMETRIC IDENTITIES. See Trigonometry Series.

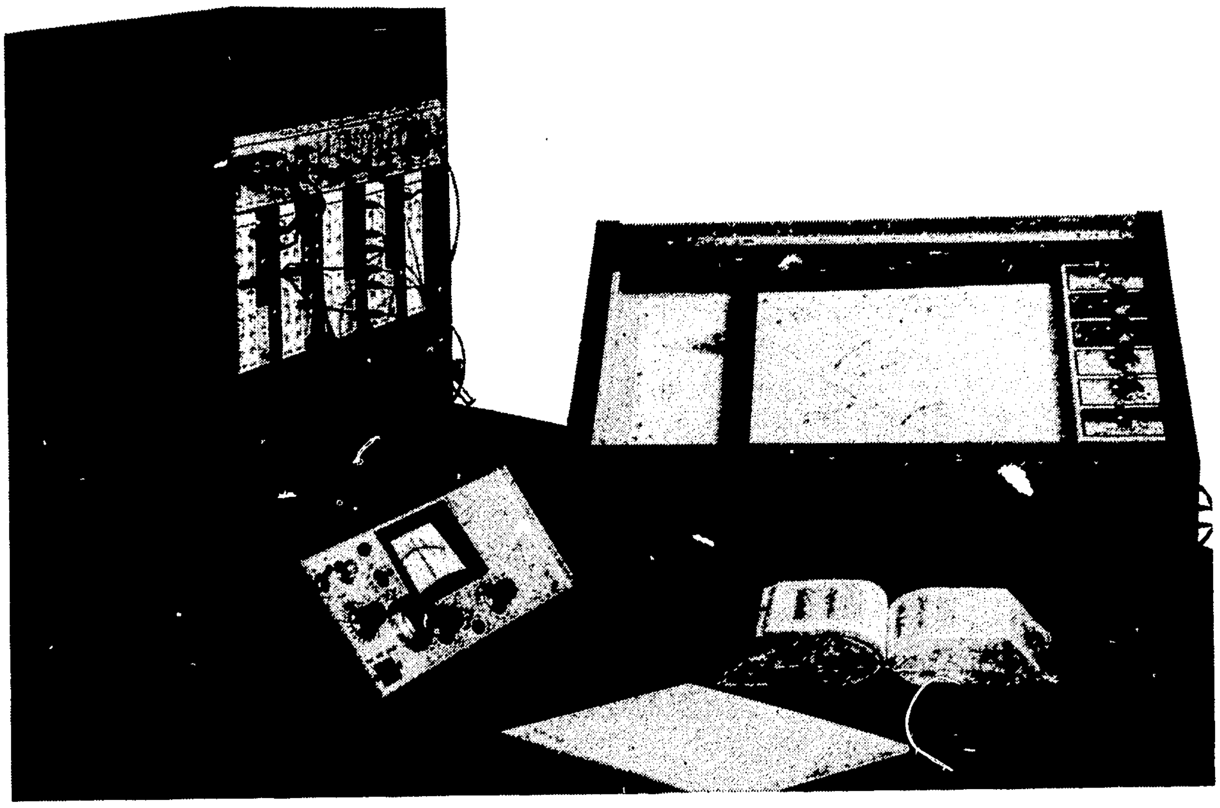


Photo by permission of Electronic Associates, Inc.

Fig. 2-1. An Electronic Analog Computer

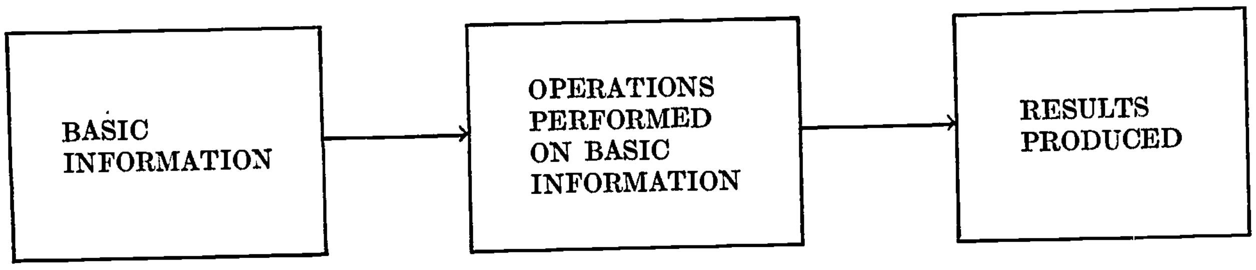
volve the representation of physical situations by mathematical equations. Large numbers of components of the type referred to above can be connected into a network by wires which can be plugged into sockets; this becomes an electrical analogy of an event such as, for example, the launching and subsequent orbiting of a satellite. Thus it is possible to produce a "mathematical" flight of the satellite in the laboratory, and thereby study the design of the system prior to actual testing.

Fig. 2-1 illustrates one type of electronic analog computer.

Digital Computers

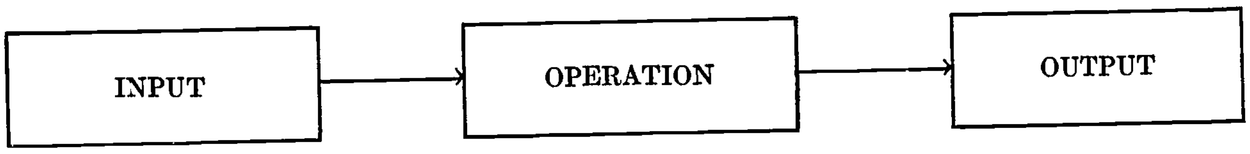
Numerous textbooks and other written materials that describe the technical operation of digital computers are available. In line with the basic philosophy of this book, this chapter will stress the mathematical concepts related to the digital computer, rather than the technical details of the computer or its operation. At times, however, it is not possible to present such material without some overlapping.

The examples given in Chapter 1 were intended to show that efficient problem solving depends on some sort of logical sequence of steps; to this extent all the examples were similar. In other words, the solution of each problem depended on (1) certain information being available at the beginning, (2) some kind of operation or operations being performed on this information, and (3) some result being produced as a result of these operations. The procedure can be represented by a diagram as follows:



Note that the arrows indicate the flow of the process.

In this simple form many processes can be described. Consider the case of a person being asked a question. It is easily seen that three basic steps are involved: (1) The basic information (spoken word) is received by the ear. (2) Certain operations are performed (in the brain) on the basic information. (3) The result (verbal answer) is given. In this sense, a human being can be considered an *information processing device*, which is a generalized name sometimes given to digital computers. For convenience, the three basic concepts involved may be expressed as *input*, *operation*, and *output*. They may also be diagrammed as follows:



Each of the operations listed below involves the concepts of input, operation and output, as illustrated by the following example:

EXAMPLE. A mathematics classroom.

Input. Teacher's verbal or written directions on the assignment received by the student.

Operation. Student performing necessary operations to do assignment.

Output. Assignment turned in to the teacher in written or verbal form by the student.

- Using a dial telephone.
- Using an adding machine.
- Using an automobile fuel gauge.
- Using a food vending machine.
- Using an electric clock.
- Using a typewriter.
- A football team on the offense.
- A combination of airplane, pilot, and control tower while the plane is approaching a landing.
- A meeting in which the chairman is listening to a number of persons discussing the two sides of a question which is finally settled by (1) the chairman's decision and (2) a vote.

A digital computer is similar in many ways to the examples and discussion given above: (1) It receives information. (2) It performs operations on the infor-

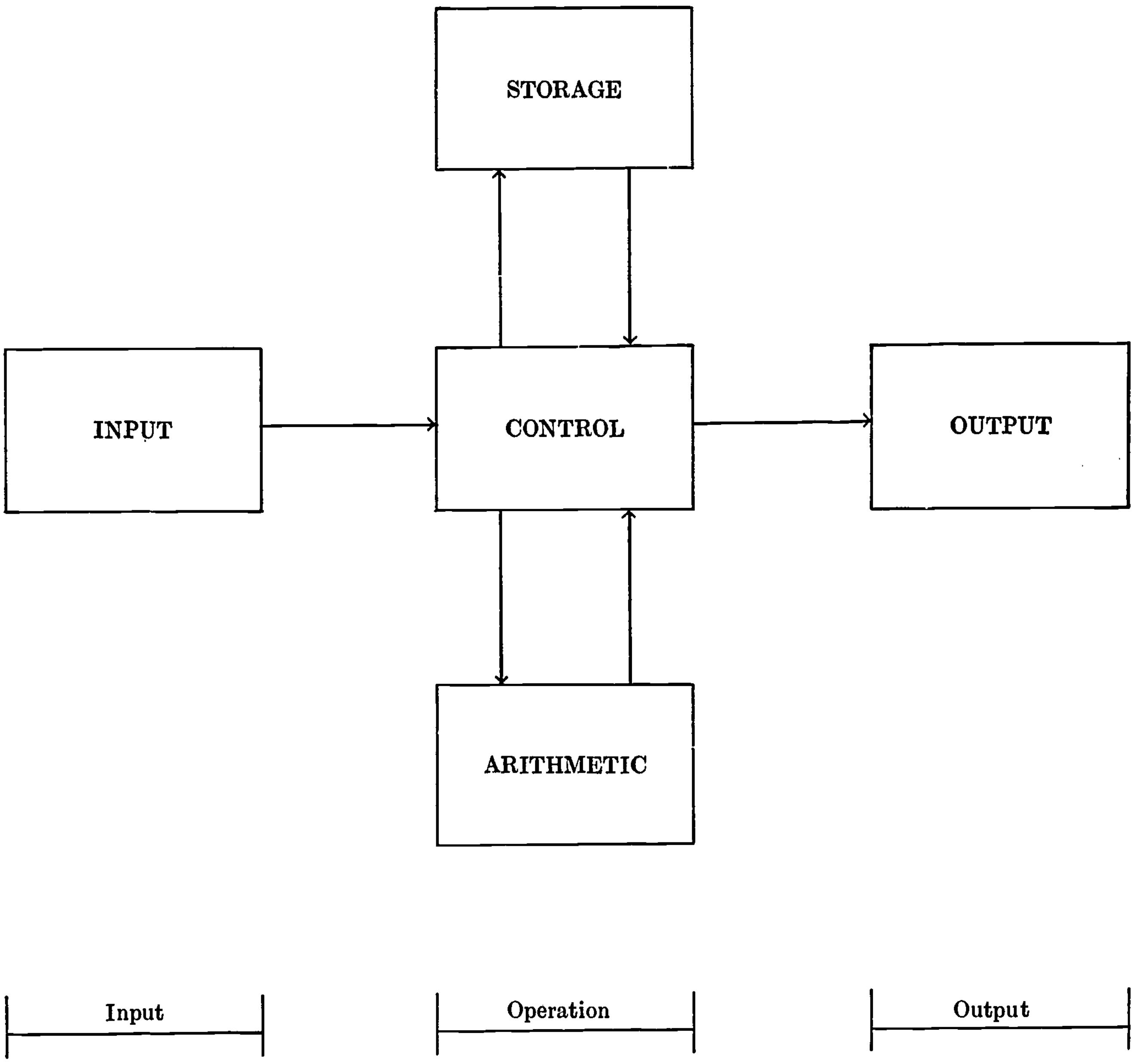


Fig. 2-2. Three-Step Device with Storage, Control, and Arithmetic Subelements

mation. (3) It produces results. By studying the idea of a three-step process (input-operation-ouput), it is possible to gain a clearer concept of the basic elements of a digital computer. The three-step diagram now has the appearance shown in Fig. 2-2.

Input places information into the computer. This may be done by means of a card with holes punched in it; paper tape with holes punched in it; magnetic tape with magnetized spots on it; or an electronic device, which automatically, by means of a radio signal, puts information into the machine.

Information may also be put into the computer by direct keyboard entry. However, the pressing of keys on a keyboard is a relatively slow process. Since most computers are unable to perform any other operations while input is under way, keyboard entry is avoided except for limited amounts of information. It is more economical to prepare cards or tape by hand, away from the computer and to enter the information at high speed through the special input devices.

A punched card, punched tape, and magnetic tape are shown in Fig. 2-3. The card and paper tape contain holes. The combination and position of the holes provide a code corresponding to numerals, letters of the alphabet, or other symbols. The magnetic tape (similar in appearance to the tape used in home or office tape recorders) does not contain holes; instead, on its surface are spots which may be magnetized. Again, the position and combination of magnetized spots correspond to various symbols. The piece of tape shown in Fig. 2-3 was sprinkled with fine iron filings to make the positions of the magnetized spots visible. In actual use these spots are not visible and are detected electronically as the tape moves across a reading element, which is similar to the one used in home tape recorders.

Information may be entered on the cards or paper tape by means of a special machine which has a typewriter keyboard. As the operator presses the keys, holes are made in the card or paper tape. The cards or paper tape are placed in the *input* device, where the holes are sensed electrically or optically with the aid of photoelectric cells. The resulting electrical impulses are then entered into the operation element and appropriate circuits are activated.

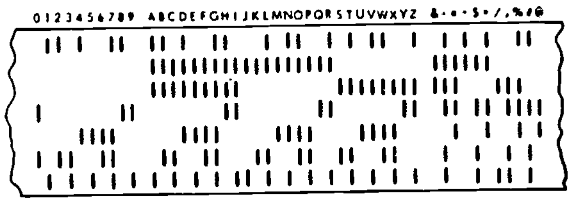
Operation does the work on the information entered into the computer through the input element. It consists of three basic units: *storage*, *control*, and *arithmetic*. The diagram shown in Fig. 2-2 may now be amplified as follows: The storage unit is used for temporarily storing information until required for some purpose; the arithmetic unit performs the usual operations (addition, subtraction, multiplication, and division) on the numbers; the control unit keeps the steps of the process going according to previously entered instructions.

Output gets the information out of the machine. It may punch holes in cards or paper tapes, record information on magnetic tape, print on paper, or even make other computers or machines do certain things.

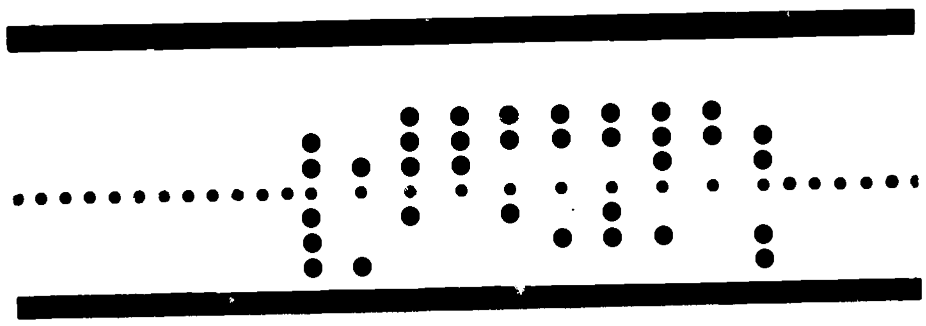
A natural question at this stage is: How does everything get started in the first place? The process starts when an individual begins to analyze the problem in a manner similar to the one demonstrated in Chapter 1. A flow diagram may be made of the process which is to be followed. After the flow diagram or other similar procedure is completed, it is necessary to enter into the machine (through *input*) the data on which operations need to be performed as well as machine instructions for carrying out the problem-solving procedure.

Essentially, this means that the human being must "communicate" with the machine, not in English or Russian or Japanese, but in still another language in such a way that the computer's capabilities for addition, subtraction, etc., will be called for in the *proper sequence*, the *proper number of times*, and in connection with the *proper data*. Since the machine can essentially do but one thing at a time, the human being may need to write thousands of instructions in the language "understood" by the machine to get his problem done. The magnitude of this task may be realized to some degree by imagining the process of giving instructions for solving simultaneous linear equations to an individual who only knows how to add, subtract, multiply, and divide, but knows no other concepts of arithmetic or mathematics and who must follow the list of instructions exactly, without benefit of questioning or inquiry.

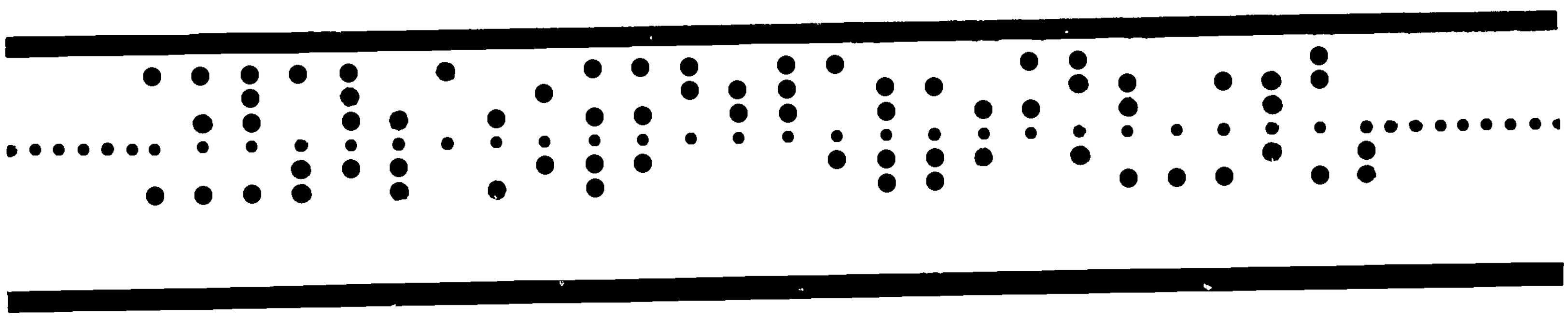
The remainder of this chapter is devoted to discussing the digital computer and techniques for using it. Part II pertains to the use of what is known as



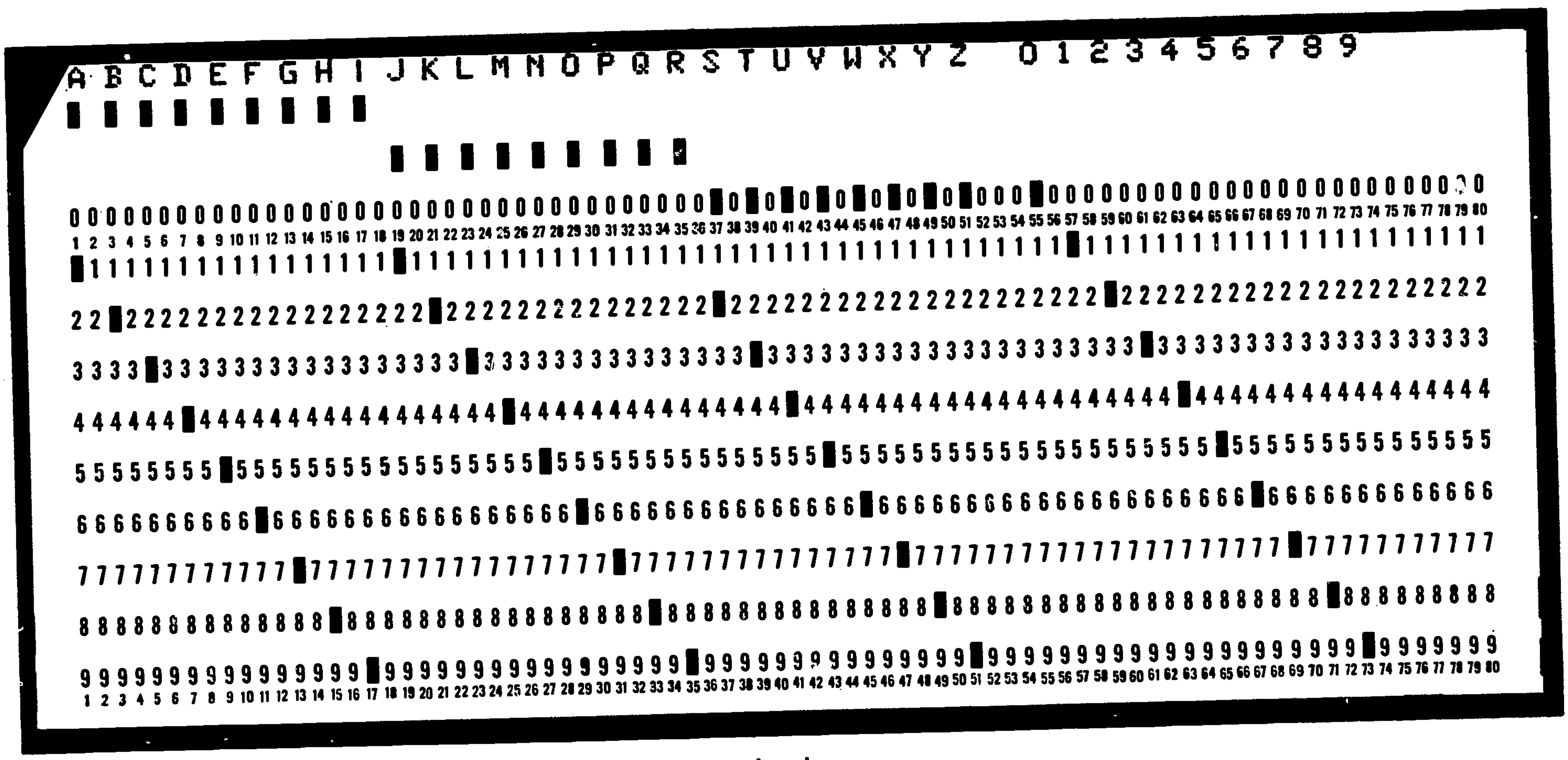
(a) Magnetic tape records information by the presence and absence of magnetized areas arranged according to code as shown in the representation above. Actual tape is one-half inch wide.



(b) Punched tape, numerical code reads left-to-right, 0.9.



(c) Punched tape, alphabetic code reads left-to-right, a-z.



(d) Punched card showing alphabetic and numerical codes.

Fig. 2-3. Media for Input Information

machine language programming. Part III deals with what is called *compiler programming*. Both techniques are similar in the sense that in either case the problem must be carefully considered and its solution planned accordingly. They are different in the sense that, once the planning is done, the compiler allows the remainder of the job to be done relatively easily.

Part II. Machine Language Programming

Detailed instructions must be given to a computer for each part of the job to be done. In some ways this parallels the procedure for solving a problem with pencil and paper or a desk calculator. Of course, there are differences as well.

The COMPIAC Computer

For illustrative purposes an imaginary computer called COMPIAC (Computer-Oriented Mathematics Project Illustrative Automatic Computer) is described. In general, it is like many computers now in operation. In detail it is, of course, unlike any specific computer. For example, the codes corresponding to the arithmetic operations, the size of numbers it can accommodate, the storage numbering systems, etc., should be considered only as being typical of real computers. Furthermore, it is a simplified computer that does not include all types of input–output devices or operation codes. A diagram of COMPIAC is shown in Fig. 2-4.

Within the computer elements of input, output, storage, arithmetic unit, and control, certain connections between electronic components are represented by arbitrary symbols. Since the machine is designed to do arithmetic, the symbols usually chosen are those of the familiar decimal notation for numbers. For example, the operation *add* may be represented by the symbol "12." The control unit is designed in such a way that connections among various components cause addition to be performed in the arithmetic unit as a result of "12" being "read," or "sensed," and decoded by the machine.

This means that the person solving a problem must somehow arrange for the control unit to have a "12" available for use when the operation *add* is required. Likewise, other symbols such as "12" must be available in sequence for all the other steps of a given problem.

A person operating a desk calculator or adding machine may have a written sequence of instructions to help him remember what he is to do, or he may have the entire sequence "stored" in his brain. He then performs the proper steps one by one. The point is that the sequence of instructions is not stored inside the desk calculator itself.

In the case of the digital computer, the sequence of operations necessary to solve a given problem is entered into it before the problem-solving process starts. The sequence of instructions is held within the storage unit which can hold many symbols. Accordingly, the entire sequence of instructions for solving a problem, represented by symbols such as the "12" mentioned above, is placed in the storage unit prior to the time the problem is to be solved. Such computers are called *stored-program computers* since all the steps in a procedure are stored before the problem-solving process is begun.

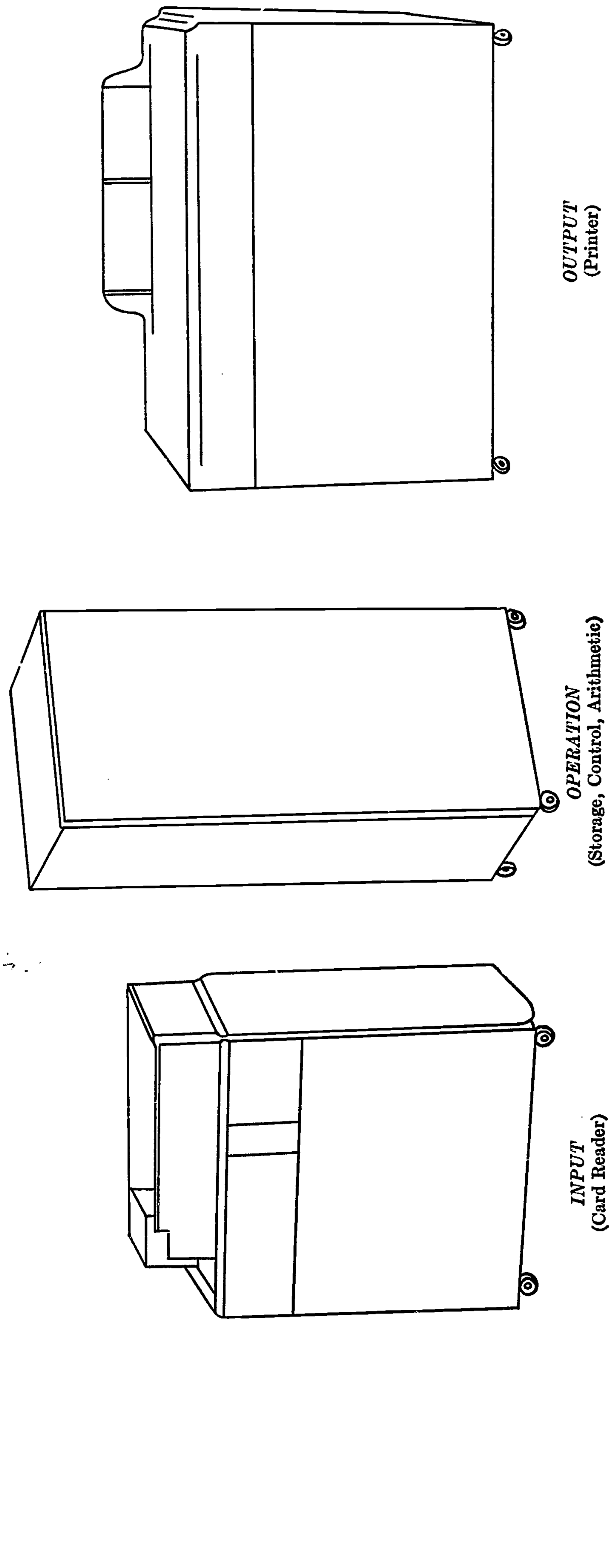


Fig. 2-4. Schematic Diagram of COMPIAC

The ability to store symbols—the stored-program concept—is one of the most important single features of the digital computer. Another important feature is its speed of operation. However, these two items, stored-program capability and speed, are closely related; that is, the speed is great, but the cost to develop and build a device which has this speed is also great. It behooves the user, then, to take full advantage of this speed. In other words, if an arithmetic operation such as addition can be performed in a few millionths of a second, it would be absurd to slow down an expensive computer by giving it something to do only every five or ten seconds—as one would operate a desk calculator.

By means of comparatively inexpensive machines, the entire sequence of instructions for a complete problem may be recorded on cards or tape. A card punching machine is shown in Fig. 2-5. These instructions are then entered into the storage unit very quickly, at one time, through the card or tape reading machines that comprise the input element. Once the entire program of instruc-

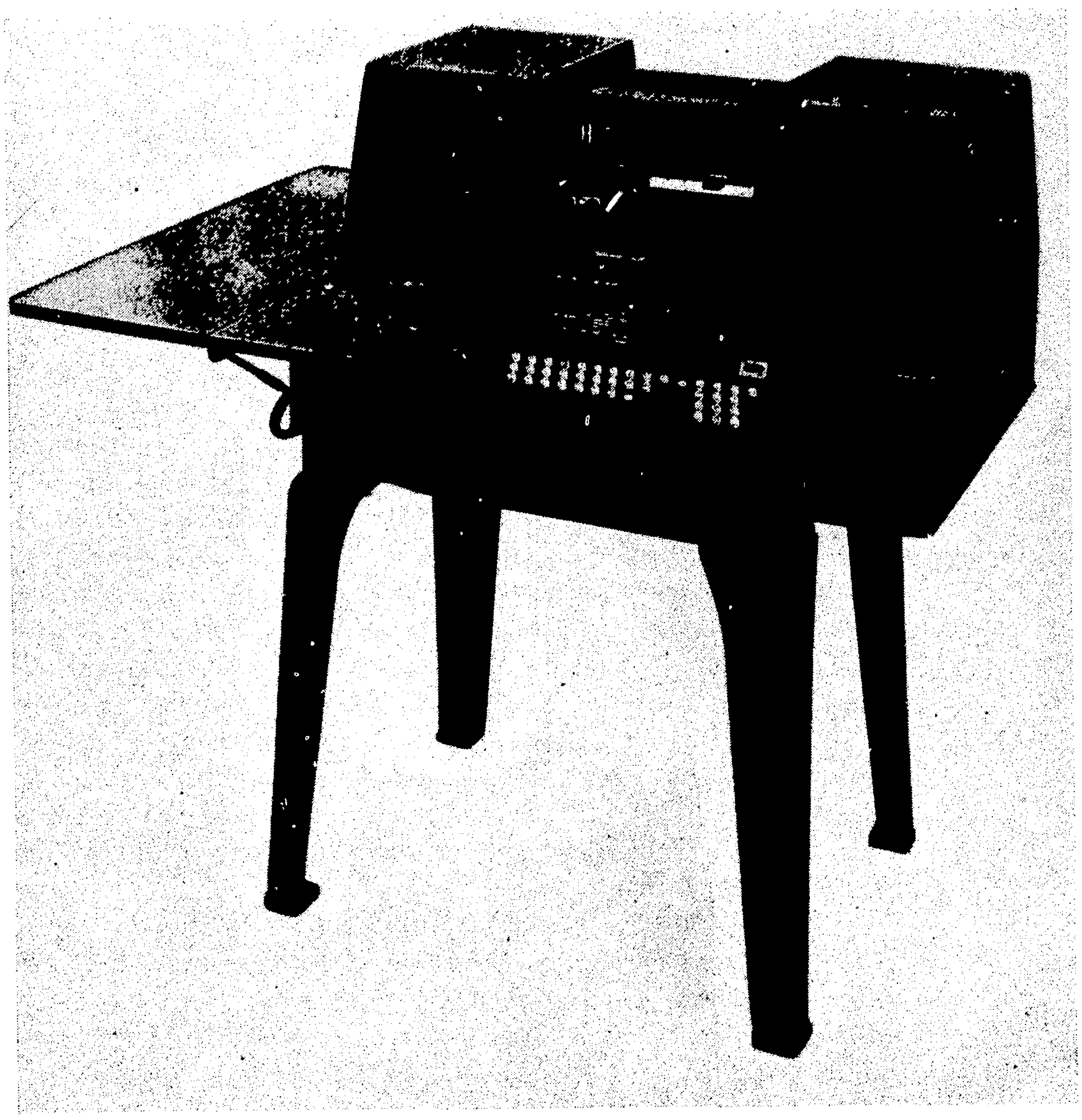


Photo by permission of the UNIVAC Division of Sperry Rand Corp.

Fig. 2-5. Card Punching Machine

tions is stored, the execution of the steps necessary to solve a given problem can follow without the delay which would occur if after each step was completed the operator had to enter the next instruction manually.

Of course, the problem cannot be solved by instructions alone—there must be some data, too. Again, to make use of the computer's speed, the data also must be in the storage unit when it is needed, since it would be just as uneconomical for the computer to wait for data as for instructions.

In general then, problem solving does not get under way until both the data and the instructions are in the storage unit—both having been entered with the aid of card- or tape-reading machines which make it possible for information to be entered rapidly into the storage unit. After all the instructions and data have been entered into storage, COMPIAC will take its first instruction from storage location 0000. It will execute the instruction found in 0000, then proceed to 0001, then 0002, etc.

It should be pointed out that as far as COMPIAC is concerned there is no characteristic difference between the symbols in storage representing data and those representing instructions. That is, the symbol "12" which represents the instruction *add* looks just the same in its part of the storage unit as the number 12 would look if it were located elsewhere to represent a piece of data. A natural question is then: Why don't they get mixed up? The answer is that they would get mixed up if the problem were not properly prepared in advance.

Most computers are designed to manipulate the symbols stored in them so that the results may be interpreted either as operations or numbers. For this reason the word *number* may seem ambiguous in the discussion which follows. However, the meaning should be clear from the context whenever the word *number* is used in reference to what more properly may be called a *symbol*.

For this discussion suppose COMPIAC has a storage unit large enough to hold two thousand numbers or two thousand instructions or some combination not exceeding two thousand; and that each number consists of six digits and its algebraic sign, + or −. If two numbers are to be added, it is necessary for the control unit to select the proper numbers from the storage unit and for the arithmetic unit to add them. Actually, the control unit must be "told" where the two numbers are stored. Accordingly each of the two thousand storage locations has what is called an *address* (analogous to the number on a seat in a theatre) which the control unit uses to select the proper storage unit position.

The addresses of the storage unit are considered to begin at 0000 and extend consecutively to 1999. The numbering depends on the design of the machine and cannot be changed by the operator just as the "address" of a theatre seat cannot be changed by a theatre patron.

If an instruction such as "12 1010" is received by the control unit, it will be decoded as meaning: Add the number stored in the location that has the address 1010 to whatever number is already in the arithmetic unit. (For brevity, the phrase "the location that has the address XXXX" is condensed simply to "address XXXX" or "location XXXX" in the following text.)

Before writing a sample program, assume that the following instructions are wired into the circuits of COMPIAC. That is, COMPIAC has been constructed in the factory to decode certain numbers and, as a result, set up particular circuits to do certain things.

The following table shows how COMPIAC would decode a given operation code and address number and carry out the instruction.

Operation code	Address number	Action
12	XXXX	*Add* the number in storage location XXXX to the number already in the arithmetic unit.
13	XXXX	*Subtract* the number in storage location XXXX from the number already in the arithmetic unit.
14		*Reset* the arithmetic unit to 0.
15	XXXX	*Put* the contents of the arithmetic unit into storage location XXXX, replacing whatever may be there, but retain the value in the arithmetic unit as well.
16	XXXX	*Stop* the computer. When the START button is pressed out, start executing operations again at address XXXX.

NOTE. The notation "XXXX" is used to symbolize any address. In a real problem, of course, numbers would replace the X's as shown below. Since the *reset* operation (code 14) does not involve any storage location, no address is given for it.

Programming for Addition

Let us suppose the problem of adding six numbers is to be solved. The following questions come to mind: How will the numbers be entered into the computer? How will the instructions be entered into the computer? What will the instructions be? How will the result be taken out of the computer? The answers to these questions are as follows:

The program planner must be careful to keep the data or numbers separate from the instructions so that they will not get mixed up. Accordingly, he may arbitrarily decide to put the data in the middle part of the storage unit since COMPIAC will automatically start its operation at location 0000.

Using a card-punching machine (shown in Fig. 2-5), the operator will punch six data cards. Some samples of data cards are shown in Fig. 2-6. These cards, when read by the input card reader (shown in Fig. 2-7), will cause the data to be entered into the storage unit; for example, the number 000067 will be stored in the location 1000, the number −000164 will be stored in location 1001, etc.

The following sequence of instructions will cause the six numbers previously stored in locations 1000, 1001, 1002, 1003, 1004, 1005 to be added and the result to be put in storage location 1999.

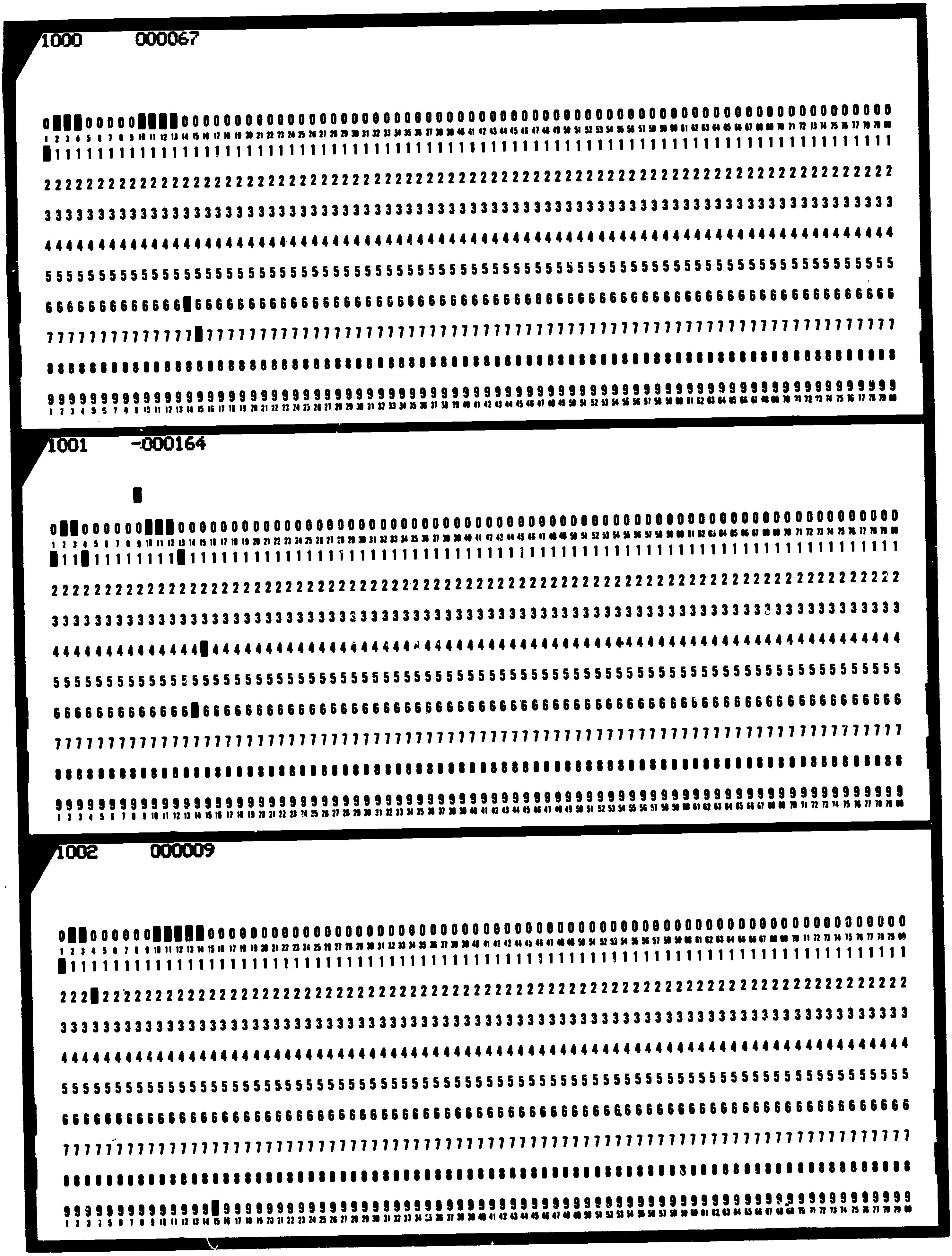


Fig. 2-6. Examples of COMPIAC Data Cards

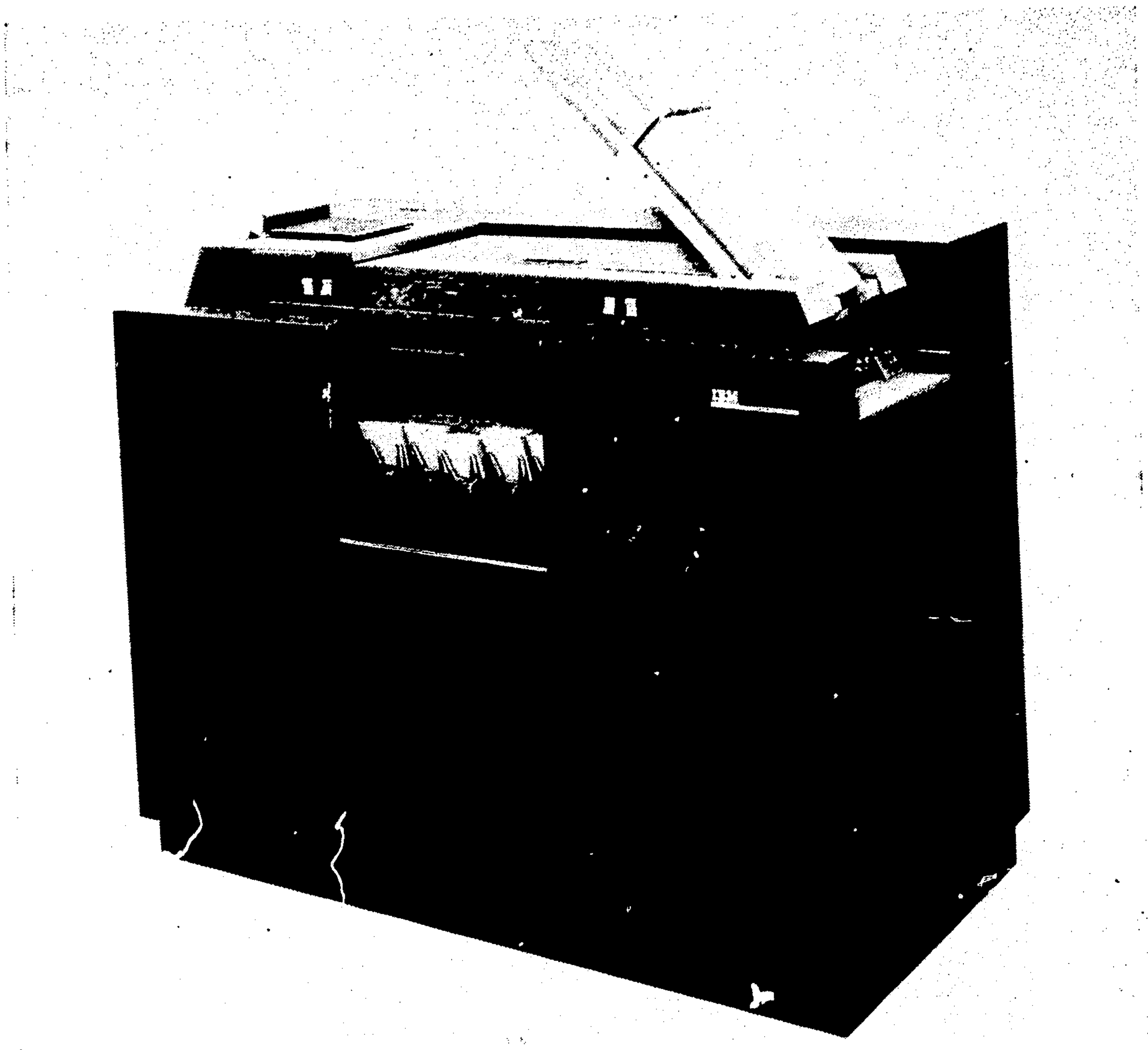


Photo by permission of International Business Machines Corp.

Fig. 2-7. Card Read Punch Machine

(This machine performs the functions of reading and punching cards.)

Storage location	Contents of the location		Meaning
	Operation	Address	
0000	14		*Reset* arithmetic unit to 0
0001	12	1000	*Add* 1st number
0002	12	1001	*Add* 2nd number
0003	12	1002	*Add* 3rd number
0004	12	1003	*Add* 4th number
0005	12	1004	*Add* 5th number
0006	12	1005	*Add* 6th number
0007	15	1999	*Put* away the result
0008	16	0000	*Stop*

Fig. 2-8 illustrates cards of the type that are used to place the above instructions into the storage unit in the same way that was used in entering the data. Although the data and the instructions are in the same numerical form, they are placed in the storage unit in such positions that they do not interfere with each other.

After the data and instruction cards have been punched, they are placed in a

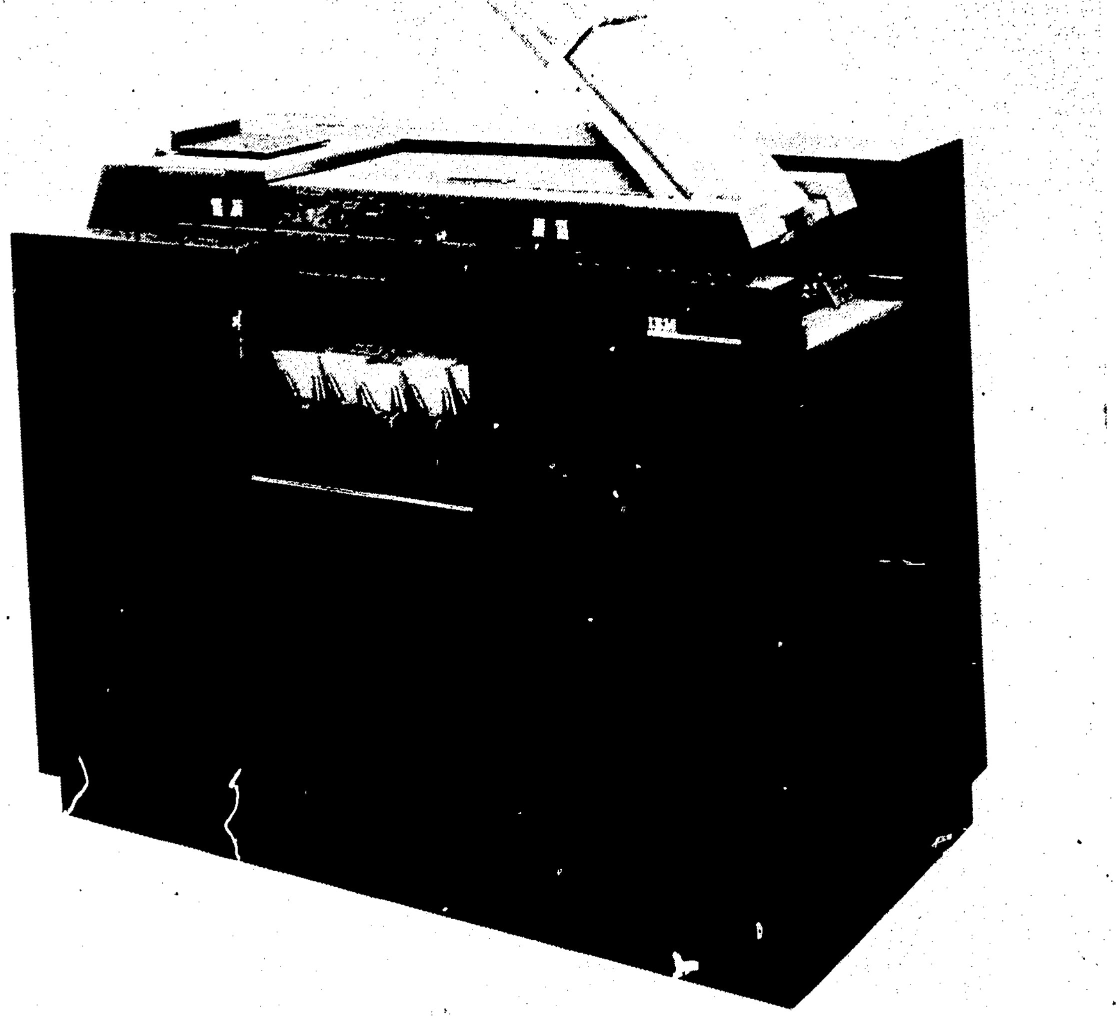


Photo by permission of International Business Machines Corp.

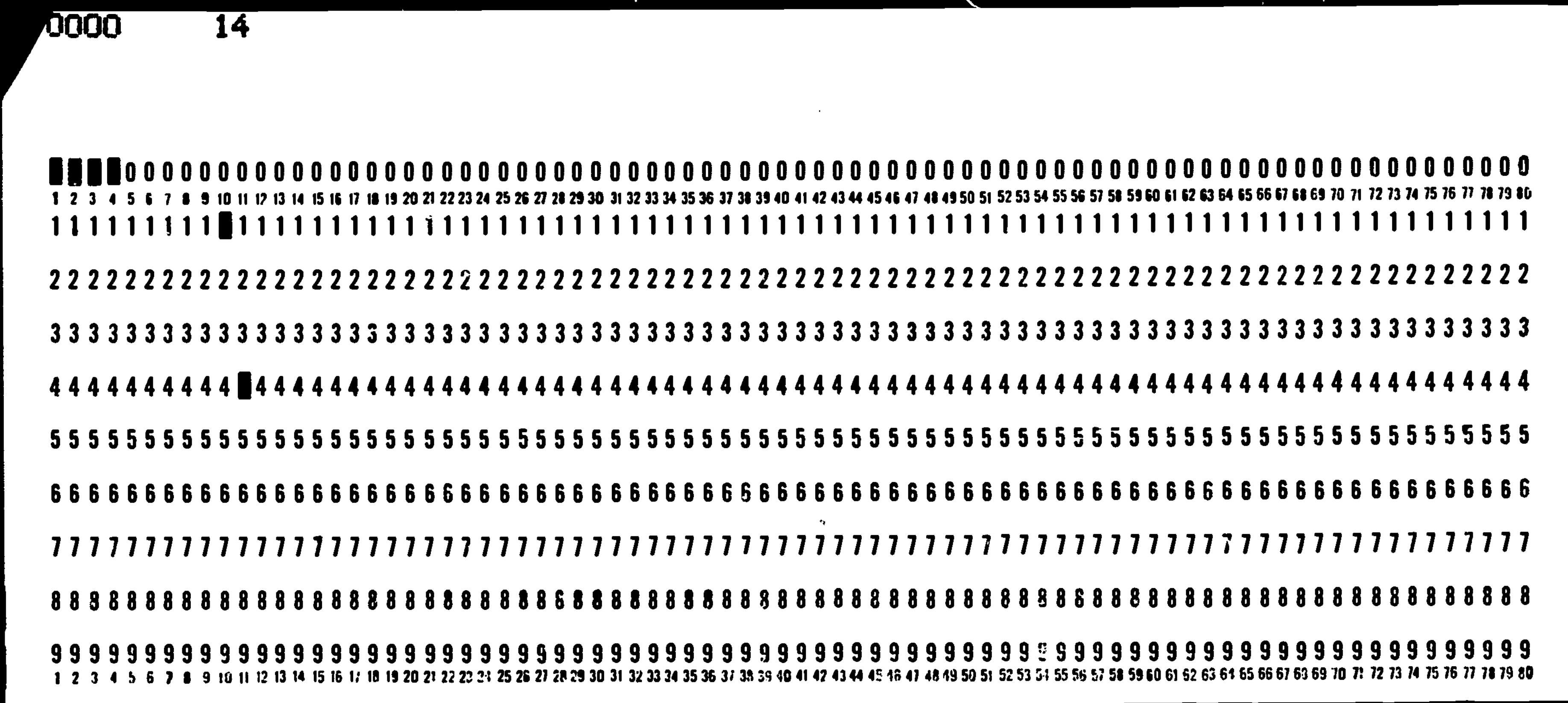
Fig. 2-7. Card Read Punch Machine

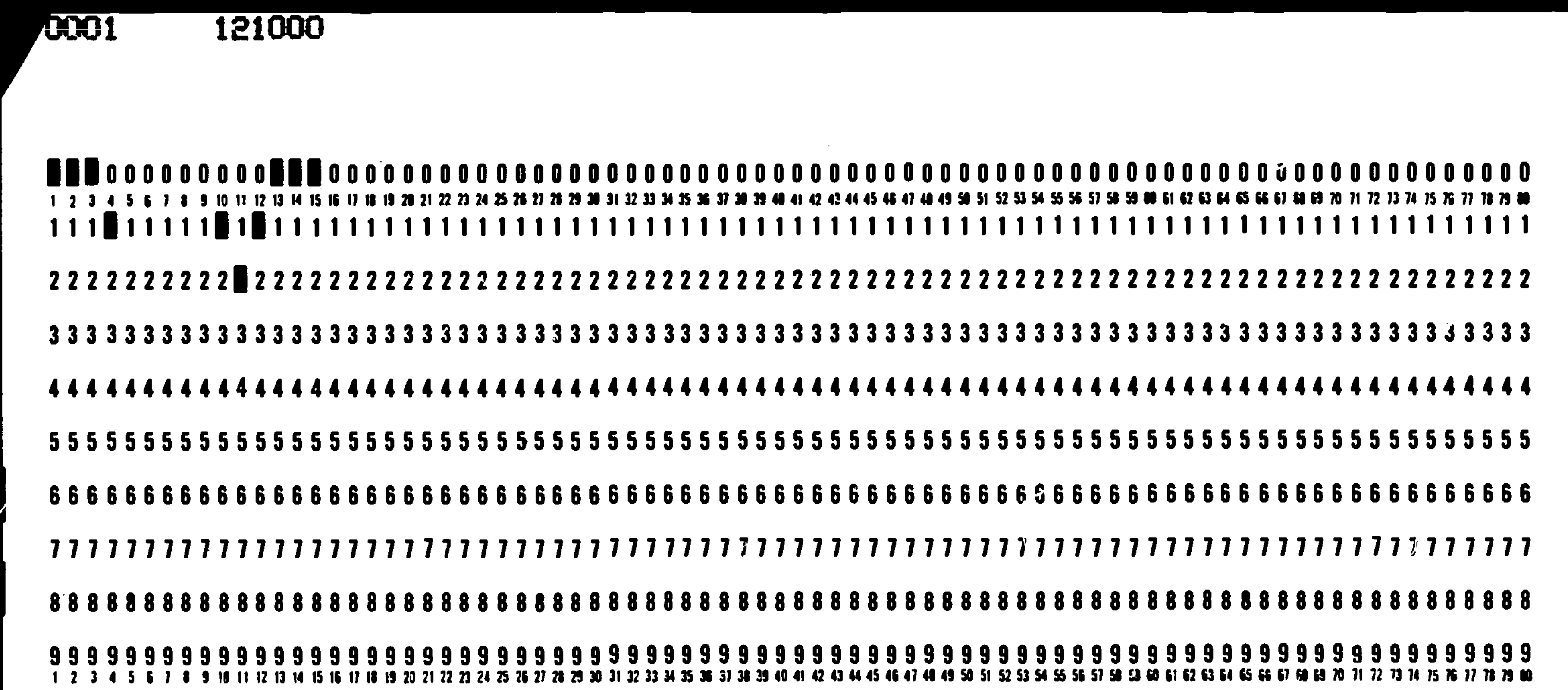
(This machine performs the functions of reading and punching cards.)

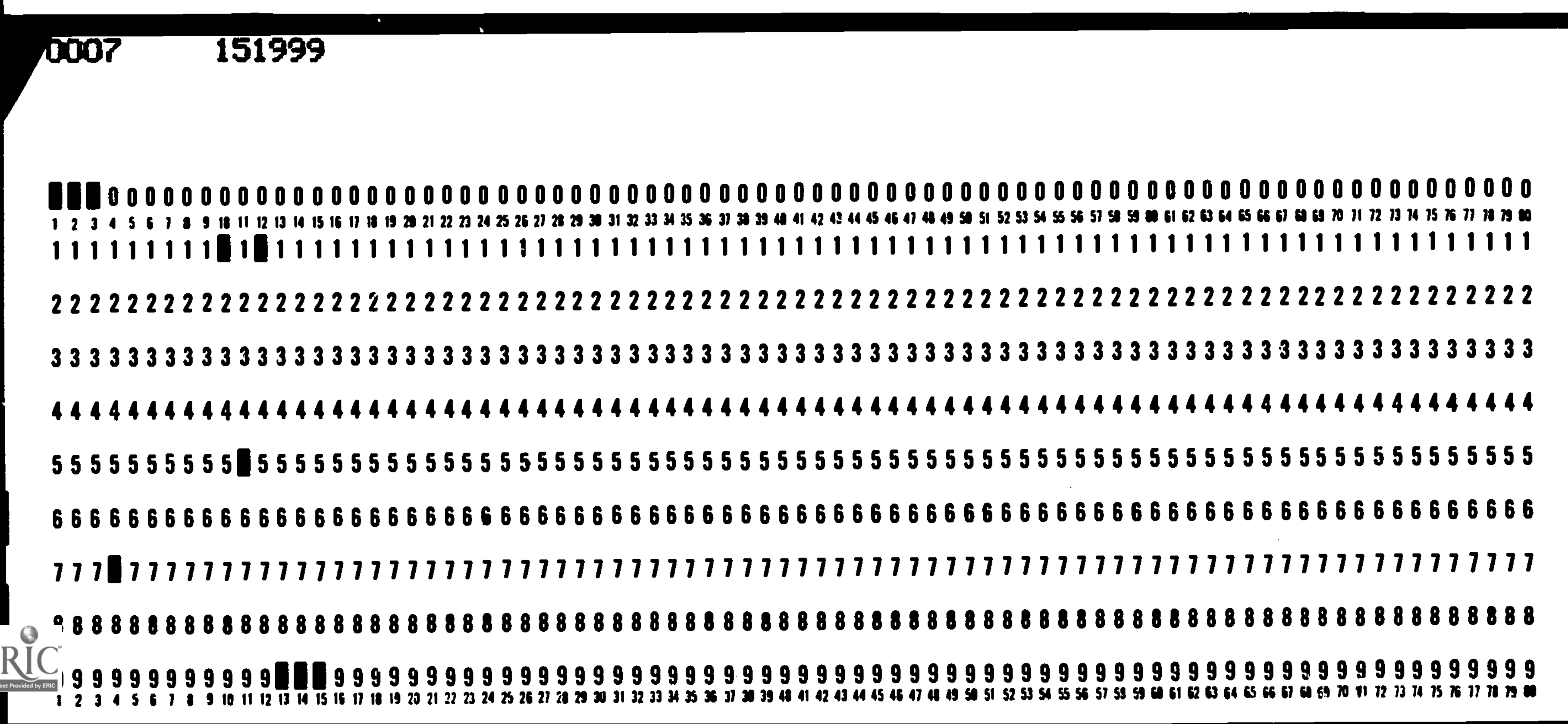
Storage location	Contents of the location		Meaning
	Operation	Address	
0000	14		*Reset* arithmetic unit to 0
0001	12	1000	*Add* 1st number
0002	12	1001	*Add* 2nd number
0003	12	1002	*Add* 3rd number
0004	12	1003	*Add* 4th number
0005	12	1004	*Add* 5th number
0006	12	1005	*Add* 6th number
0007	15	1999	*Put* away the result
0008	16	0000	*Stop*

Fig. 2-8 illustrates cards of the type that are used to place the above instructions into the storage unit in the same way that was used in entering the data. Although the data and the instructions are in the same numerical form, they are placed in the storage unit in such positions that they do not interfere with each other.

After the data and instruction cards have been punched, they are placed in a


0000 14


0001 121000


0007 151999

single pile and put into the hopper of the card reader, ready for the process to begin. Since each card is identified by its storage location number, the order of the cards is not important. When the operator is ready to begin, the START button on the console of COMPIAC is pressed. This causes the cards to be read in—one at a time, but without interruption—until all have been read. After the last card has been read, the control unit initiates the problem-solving process by first executing the instruction in address 0000 and continuing automatically until the *stop* instruction is reached.

Self-Modified Instructions

The operation part of the instructions ("12") is the same in all of the steps 0001 through 0006. Notice also that the corresponding addresses increase by 1 from 1000 through 1005.

The instruction in location 0001 (12 1000) could be used again for the second number and do the same job as the instruction in 0002, if the number 1000 could be somehow changed to 1001. And if this change were made, the same instruction could be used once more for the third step, if the number 1001 could be changed to 1002.

The stored-program computer can be made to change its own instructions; this is one of its greatest advantages. If properly programmed, it can execute an instruction such as the 12 1000 in location 0001; then, the machine will itself add 1 to the instruction, making it now 12 1001; then, the machine will execute this new self-modified instruction; and, then, modify and execute the next and so on in accordance with the program given the machine.

If one hundred numbers were to be added instead of six, the above listing of instructions would have 103 steps. This would be tedious to write. Also, 103 positions in storage would be required for all the necessary steps, instead of the nine required above.

Some of these undesirable features can be eliminated if the computer has certain other instructions available, as follows:

Operation code	Address number	Action
17	XXXX	Execute as the next instruction whatever is found in storage location XXXX and then continue, in sequence.

This instruction then can interrupt the step-by-step sequence and direct it to begin in a new place.

The following additional instruction is also needed to allow self-modification of instructions:

Operation code	Address number	Action
18	XXXX	If the number in the arithmetic unit is *positive*, execute as the next instruction whatever is found in storage location XXXX and then continue from that point. If the number is *zero*, proceed in sequence.

With the aid of these instructions it is possible to produce repetitive cycles, or loops, and thus perform the self-modifying steps necessary to add 100 numbers without using 103 steps. The general procedure is to have the computer perform the following operations: (1) add 1 to the addresses as necessary to go from step to step; (2) keep track of how many times this has been done so that the cycling can be stopped after 100 times. Since the computer must have something to use in changing the addresses from step to step, the number 1 must be available in some storage location. The number 100 must also be available so that in each cycle, by subtracting 1 to obtain 99, then 98, then 97, etc., the computer will be able to stop the cyclic process (with instruction code 18) when the subtractions finally reduce the 100 to 0.

One additional instruction is necessary for the output printer of COMPIAC to print the result:

Operation code	Address number	Action
19	XXXX	*Print* the number, which is in storage location XXXX, on the output printer.

Fig. 2-9 illustrates a typical printer.

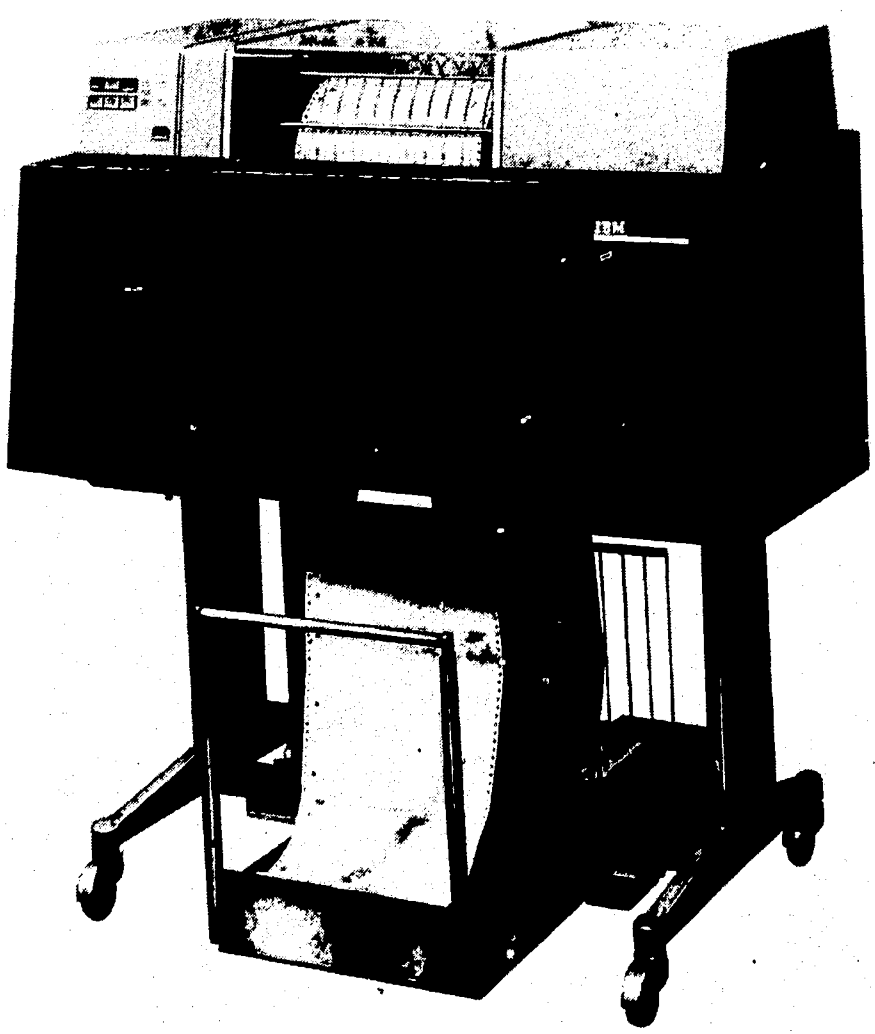


Photo by permission of International Business Machines Corp.

Fig. 2-9. Printing Machine

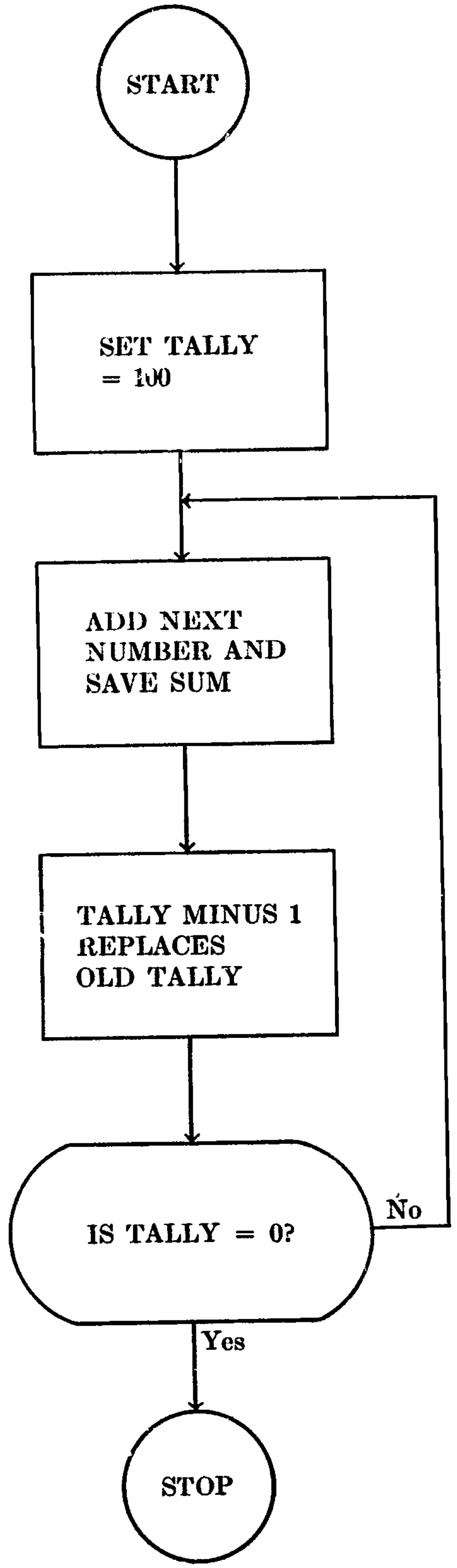


Fig. 2-10. Keeping a Tally

Instruction Address	Operation Code	Data Address	Comments
0000	14		Reset the arithmetic unit to zero
0001	12	1000	Add "next" number.
0002	15	1999	Save partial sum in location 1999.
0003	14		Reset the arithmetic unit to zero.
0004	12	0101	Add the number (tally) currently in location 0101.
0005	13	0100	Subtract 1 from tally.
0006	18	0008	If (+) go to 0008 to save new value of tally.
0007	17	0016	Go to 0016 because sum is now complete.
0008	15	0101	Replace old value of tally with new value.
0009	14		Reset the arithmetic unit to zero.
0010	12	0001	Add the instruction in location 0001.
0011	12	0100	Increase the data address of the instruction by 1.
0012	15	0001	Put modified instruction back in location 0001.
0013	14		Reset the arithmetic unit to zero.
0014	12	1999	Add partial sum.
0015	17	0001	Return to 0001 to add "next" number.
0016	19	1999	Print completed sum, which is in location 1999.
0017	16	0000	Stop. Begin again at 0000 if START button pushed.
0100	00	0001	The constant "1" required by the program.
0101	00	0100	Initial value for the tally.

Fig. 2-11. COMPIAC Coding Sheet

Tally and Instructions

Now assume that the 100 data numbers are stored in locations 1000, 1001, 1002 . . ., 1099; that the sum will be stored in 1999; and that the number 1 (a stored constant to be used for changing the addresses) is stored in location 0100. Location 0101 will initially contain the number 100 from which 1 is repeatedly subtracted to keep track of the number of times the cycle has been completed. This method of controlling a loop by repeatedly subtracting 1 until a number has been reduced to zero is one way of keeping a tally. Another way of keeping a tally is to start with 0 and repeatedly add 1 until a specified value is reached. Fig. 2-10 shows graphically the logic of the subtraction method. Fig. 2-11 shows how the instructions for the key punch operator would be written on a standard form. It should be noted that the comments are not part of the program used by the computer, but are actually notes made by the programmer for his own use.

Notice that the steps listed on the coding sheet might be written in a flow diagram form and that they do not always follow in sequence. For example, step 6 may be followed either by step 7 or step 8, depending on the value of the tally. Also, step 15 is followed by step 1 as the problem proceeds, and the process is finally terminated by a jump from step 7 to step 16 and then step 17.

NOTE. Some teaching aids for introducing the concepts discussed in the foregoing text are described in Appendix B.

References

The reader who is interested in further detail on the subject of machine language programming should consult the following references, which are identified in the Bibliography, Appendix C:

5, 9, 10, 22, 26, 44, 45, A5, A8, O3, P4.

EXERCISES

1. Copy the form shown in Fig. 2-12 and use it to follow through the example in Fig. 2-11. First, complete writing the instructions in the proper addresses; then begin at 0000, simulating the functions of the control and arithmetic units by decoding and executing each instruction until the tally in location 0101 has been reduced to 000093. If this is done on a blackboard where erasing is easy, the arithmetic unit can be represented by a box and its contents changed as the process proceeds. On paper it may be more practical to represent the arithmetic unit as shown in Fig. 2-12 by writing the successive numbers in a long row and indicating *reset* by drawing a line through the "erased" values. This will also give a record of the process.

2. (a) What is the value stored in location 1999 when the tally in location 0101 is reduced to 000093? (b) What are the contents of location 0001 at the time the tally is reduced to 000093? (c) The program will not add the same numbers again if it is restarted at location 0000. Why? (d) Modify the program so

STORAGE

INSTRUCTION ADDRESS → / OP CODE → DATA ADDRESS					
	0000 14	0001 121000	0002 151999	0003 14	0004 120101
	0005 130100	0006 180008	0007 170016	0008 150101	0009 14
	0010 120001	0011 120100	0012 150001	0013 14	0014 121999
	0015 170001	0016 191999	0017 160000	0018	0019

ARITHMETIC UNIT

0
+67
~~-67~~
0
100
-1
~~-99~~
0
121000
+1
~~121001~~
0
+67
-164
-97

STORED CONSTANT | TALLY

ADDRESS →					
	0100 000001	0101 000100			

TALLY

0101
000100
000099

DATA ADDRESS → / DATA →					
	1000 000067	1001 −000164	1002 000009	1003 −220015	1004 002600
	1005 171005	1006 121002	1007 131003	1008 181005	1009 191008

SUM LOCATION

1999
~~-67~~
-97

Fig. 2-12. Computer Simulation Form

that it will work correctly the second time through; that is, when it is restarted at location 0000.

3. (a) Can the data stored in location 1005 be interpreted as an instruction? (b) Can COMPIAC distinguish between instructions and data? (c) Assume that through a coding error the control unit is directed to execute the "instruction" at 1005. What will happen? (d) If through error the control unit executes the

"instruction" at location 1006, what are the two possible courses COMPIAC can follow?

4. (a) What is the greatest possible number of digits in the sum of one-hundred 4-digit numbers?; 6-digit numbers? (b) How many digits must the arithmetic unit hold to retain the digits developed in (a)?

5. The following problems can be flow-diagrammed and programmed with the few COMPIAC machine language instructions presented. (Of course, computers have *multiply, divide, shift* and many other instructions, but these quickly lead to complications that preferably should be studied later, in the references suggested for this chapter.) (a) Count the number of non-zero integers stored in locations 0100–0199, inclusive, and put this count in location 0200. (b) Select the greatest number stored in locations 1350 through 1399 and store this greatest number in location 0850. (c) Choose locations for a, b, c, and d and code a program to evaluate $x = a + b - 2c - d$. (d) Code a program to examine a, b, c, and d and change the sign of all positive values. Then store any changed values in the original locations. (e) A series of numbers is stored consecutively starting at location 0100. Write a program to add twenty values in 0100, 0101, 0103, 0106, 0110, 0115, . . . and print the result. (Notice that the differences between successive pairs of addresses change according to a pattern.)

ANSWERS

2. (a) 74,504. (b) 121006. (c) The tally has been reduced and the address of the data incremented. These instructions and constants would have to be reset to their initial values for the program to run again. (d) Store the initial contents of locations 0001 and 0101 in two other locations also, say 0102 and 0103, and add some instructions following location 0017 that will put these numbers into locations 0001 and 0101 before control returns to execute instruction 0000.

3. (a) Yes. It has legal operation code and address digits. (b) No, it cannot. (c) The operation code of 17 will cause COMPIAC to transfer control or "jump" to 1005, the same location. Thus the poor machine is hopelessly caught in a one-instruction loop and cannot get out of it without human assistance. (d) COMPIAC will unsuspectingly proceed to add and subtract; at location 1008 it will decide to return to the loop at location 1005 or else print the number 181005. The choice depends upon the number that happened to be in the arithmetic unit just prior to the error.

4. (a) Six digits; eight digits. (b) Eight digits. Most computers have an arithmetic unit that holds twice as many digits as a storage location, so COMPIAC's arithmetic unit would hold 12 digits as well as the sign.

Part III. Compiler Language Programming

The type of coding discussed in Part II of this chapter may become quite complicated for three reasons:

1. It is necessary to remember several numerical codes that stand for the arithmetic operations.

2. It is necessary to resort to some unusual techniques (in the "counting 100" part of the procedure) to keep track of what is happening.

3. It is difficult to correct a list of instructions either by adding a new instruction to, or removing an instruction from a list of steps since this disturbs the sequence that the computer must follow. For example, there is no convenient way of inserting an instruction between the two instructions located in positions 1010 and 1011—and still keeping the steps in sequence—except by renumbering all the instructions that follow the place in the sequence where the change was made. The same problem occurs if an instruction is removed.

Fortunately many improvements have been made during the past few years in developing so-called *languages* for use with computers. The example given in Part II of this chapter can be described as having been written in *machine language*. In other words, codes (such as 12 for addition) that correspond to the design of the computer are used by the human programmer. This places a burden of translation on the programmer. That is, he must continually translate his own thoughts involving adding, printing, etc., into machine language.

Translation by the Computer: The Compiler

Because of the problems presented by machine language it became necessary to find ways in which the computer itself could perform the translation. This would relieve the programmer of this task and give him more time for the mathematical formulation of the problems at hand. An ideal solution would be to make it possible for the programmer somehow to "talk" to the computer in the language of mathematics, thus avoiding the problem of translation.

The following pages will discuss some translating techniques that have been developed and, in particular, a device known as the *compiler*. This is a "language" consisting of a sequence of instructions, prepared by the methods that were used in the example given in Part II. The compiler is therefore a sequence of instructions to solve a special kind of problem—that of translating a mathematical language used by a human being into the machine language "understood" by the computer.

The development and improvement of compilers has attracted much attention among professional computer users. The construction of these languages is a complex and expensive job, and many groups, some on an international level, are actively working in this field. The most prominent current international effort revolves around a language called ALGOL (Algorithmic Language).

Although the preparation of a compiler requires a great deal of effort, it needs to be done only once for a given type of computer. Thereafter, all other problems can be prepared in a more convenient language. The preparation of a compiler is therefore worth the investment.

The rest of this chapter will describe the compiler and show how it is used in actual practice. For illustrative purposes an imaginary COMPIAC compiler will be discussed. It is similar to compilers now used in the United States and other parts of the world, although it may be different in detail. It is patterned after NELIAC, one of several compilers that are similar to ALGOL, mentioned above. The differences between the COMPIAC compiler and NELIAC are minor and appropriate to the objective of this discussion: to show, in a few pages, how a compiler can be used, without undue emphasis on fine points that may be beyond the interest of the general reader. It is intended here to point out some of the differences between the type of problem presentation and coding described in Part II of this chapter and the type of problem preparation and coding used in most computation laboratories, including those that do technical work as well as those that are concerned with fiscal or other management-type work.

Compiler languages, such as the one to be described—and there are several of them—do not relieve human beings from the task of thinking carefully about a mathematical or payroll or automation job. The language merely makes it easier to prepare instructions for the computer *after* the planning phases have been completed.

The type of coding discussed in Part II of this chapter depended on the symbols 0, 1, 2, · · · 9. That is, 12 was a code for what is otherwise symbolized by +, the addition sign. Similarly, 13 meant subtract, and so forth.

The important point is that if some symbol, such as 12, can be used for instructing the computer to perform the operation called addition, why not develop symbols that are more easily associated with the required operations? That is, since all symbols are similar in the sense that on paper tape or cards they are represented by certain arrangements of holes, a system is needed whereby the human being can use whatever symbols he wishes and make the compiler translate them.

Accordingly, then, the compiler about to be described, like all now in common use, can distinguish the letters of the alphabet, the numbers 0 through 9, and many special characters such as punctuation marks, parentheses, arithmetic signs, and others.

It is worth emphasizing again that the compiler is itself a *computer program* originally prepared in machine language. (Actually, only part of a compiler must be written in machine language. After the basic translation of symbols is done, in machine language, the remainder of the compiler can be compiled by itself. That is, after the basic part is done, the compiler can "lift itself by its own bootstraps.") As a program, the compiler consists of many instructions that can interpret symbols, such as 21, 6, ?, (62), A, +, $(x = 2)$, singly or in combination and according to certain rules, translate thoughts expressed by these symbols into the *numerical language* of the computer—the same type of numerical language that the human being would otherwise have to write himself.

Some compilers used with large computers (the compiler discussed here is typical) produce from five to ten machine instructions (like those in the previous example) for each compiler instruction, at a rate of 500 to 1000 per minute. In other words, a programmer who prepares a problem in compiler language and uses 50 compiler statements (such as those shown in Fig. 2-18) would otherwise need to prepare from 250 to 500 machine instructions. A specific instance of this economy is seen in coding the cyclic operations in the example where it was necessary to "count from 100." It took several instructions to get this simple job done. The compiler can operate on a single "human-being-type" instruction meaning "Do this operation 100 times, changing the addresses as necessary," and can produce all the individual instructions itself.

Programming for a Compiler

In general, the program that the human being prepares for translation by the compiler consists of two parts. These are called: (1) the *names* and (2) the *statements.*

The names are merely symbols used in solving the problem. For example, a name may be the word, NUMBER. The compiler will interpret this word—which is merely a particular set of symbols—as designating some part of the problem to be solved. (This is similar in concept to a teacher telling a class, "Take a number." The class realizes that the word *number* will be referred to again; it is a part of the problem about to be described.) The compiler, upon encountering this word, will use it to designate a particular storage location. Thus the program writer needs no longer to worry about, for example, a specific storage location, such as 1062. He needs only to remember that a storage location called NUMBER is available. Whether the storage location is really 1062 or 1063 or some other is up to the compiler. The compiler will keep track of the "bookkeeping."

Sometimes more than one number must be considered in a problem. Thus, the statement

NUMBER (100)

means that 100 locations will be specified in the problem at hand. Accordingly, when the card shown in Fig. 2-13 is read by the card reader, the compiler will set aside, in a manner similar to the problem above, 100 storage locations.

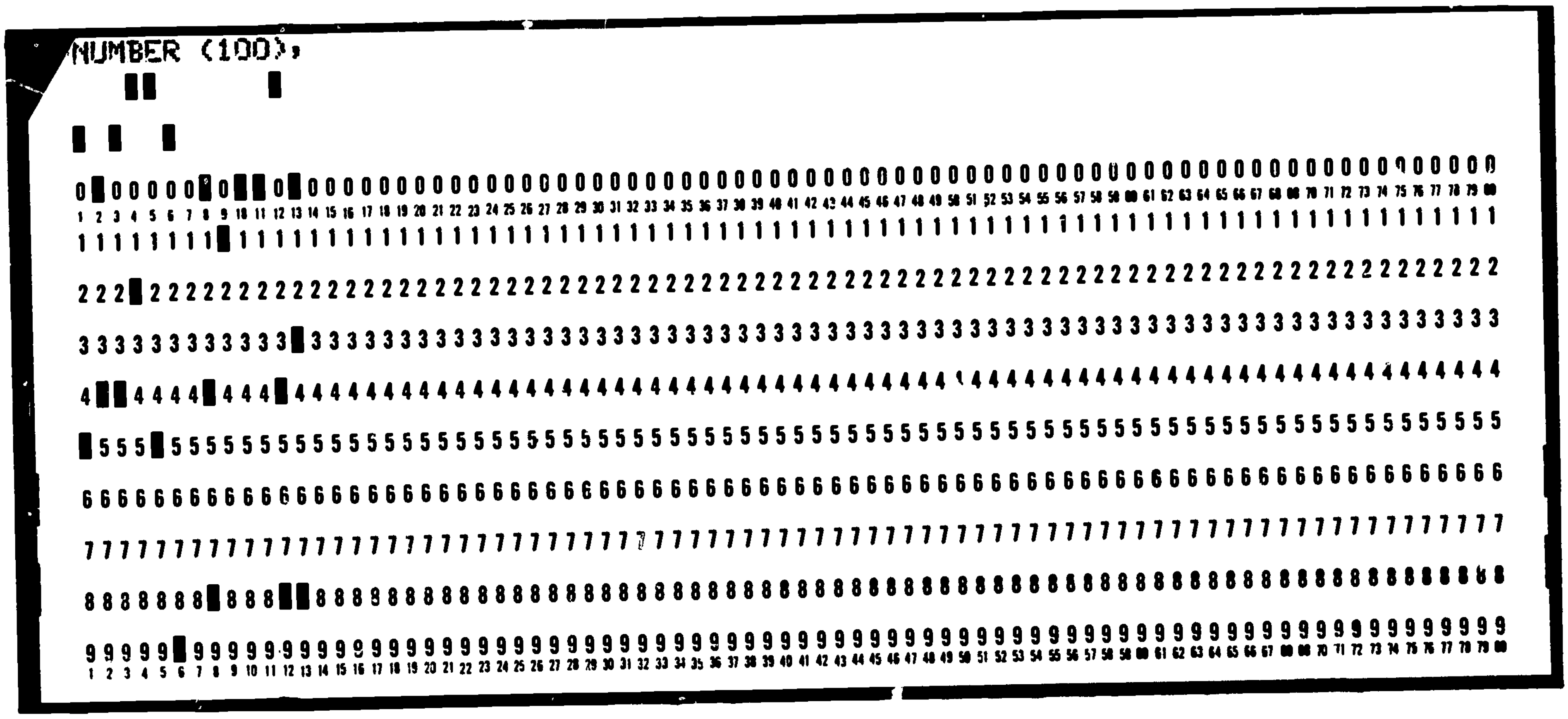


Fig. 2-13. Compiler Name Card: *Set aside* 100 *storage locations.*

A possible problem involves 100 students, each one of whom had his height, age, and weight recorded. The following names may be used:

HEIGHT (100),
AGE (100),
WEIGHT (100).

For each of the three items the compiler would assign 100 storage spaces and keep track of them. "Keeping track" means that, whenever a reference is made to one of the storage positions identified by the word HEIGHT, the compiler would not mistake it for any other position such as one of the group identified by the word AGE. It is the same idea as, in the example, assigning the 100 values for height to storage positions 1000, 1001, 1002 . . . 1099, the 100 values for age to 1100, 1101, 1102, . . . 1199, and the 100 values for weight to 1200, 1201, . . . 1299. The difference is that there are no actual storage location numbers to worry about. The compiler will allow references to the 100 locations called HEIGHT equivalent to: "Add the contents of the first (or seventh, or seventy-sixth) location in the group of 100 called HEIGHT."

An Addition Program

The following paragraphs show how a program for adding six numbers and storing the answer may be written.

First, the names:

NUMBER (6),
ANSWER (1);

instruct the compiler to set aside six locations, which will be referred to in the program as NUMBER, and to set aside one location referred to as ANSWER.

NOTE. The name ANSWER (1), listed above, can be written simply as ANSWER, it being unnecessary to designate a single location. That is, the compiler processes a name as if it pertained to *one* location *unless otherwise specified.* The use of the (1), however, does not cause difficulty—it is merely redundant. The same applies to the use of names in statements that make up the program.

In reading names, the compiler interprets a *comma* as meaning: "There is one more name in the list." The *semicolon,* on the other hand, is understood to mean: "There are no more names; the next item will be part of the program."

Second, the statement:

NUMBER [1] + NUMBER [2] + NUMBER [3] + NUMBER [4] + NUMBER [5] + NUMBER [6] → ANSWER [1].

This means that whatever is in the first location called NUMBER is added to whatever is in the second location called NUMBER, etc., and the sum is finally placed in the place called ANSWER.

NOTE. In an array of numbers, it is advantageous to refer to the first number as 0; the second, as 1; the third, as 2; etc. In attempting to describe the compiler without introducing too many details, this text refers to the first number of an array as 1; the second, as 2; etc.

The *perio* means: "This is the end of the statement." The symbol →, of course, is just another symbol such as a comma, 6, or A. It is understood by the

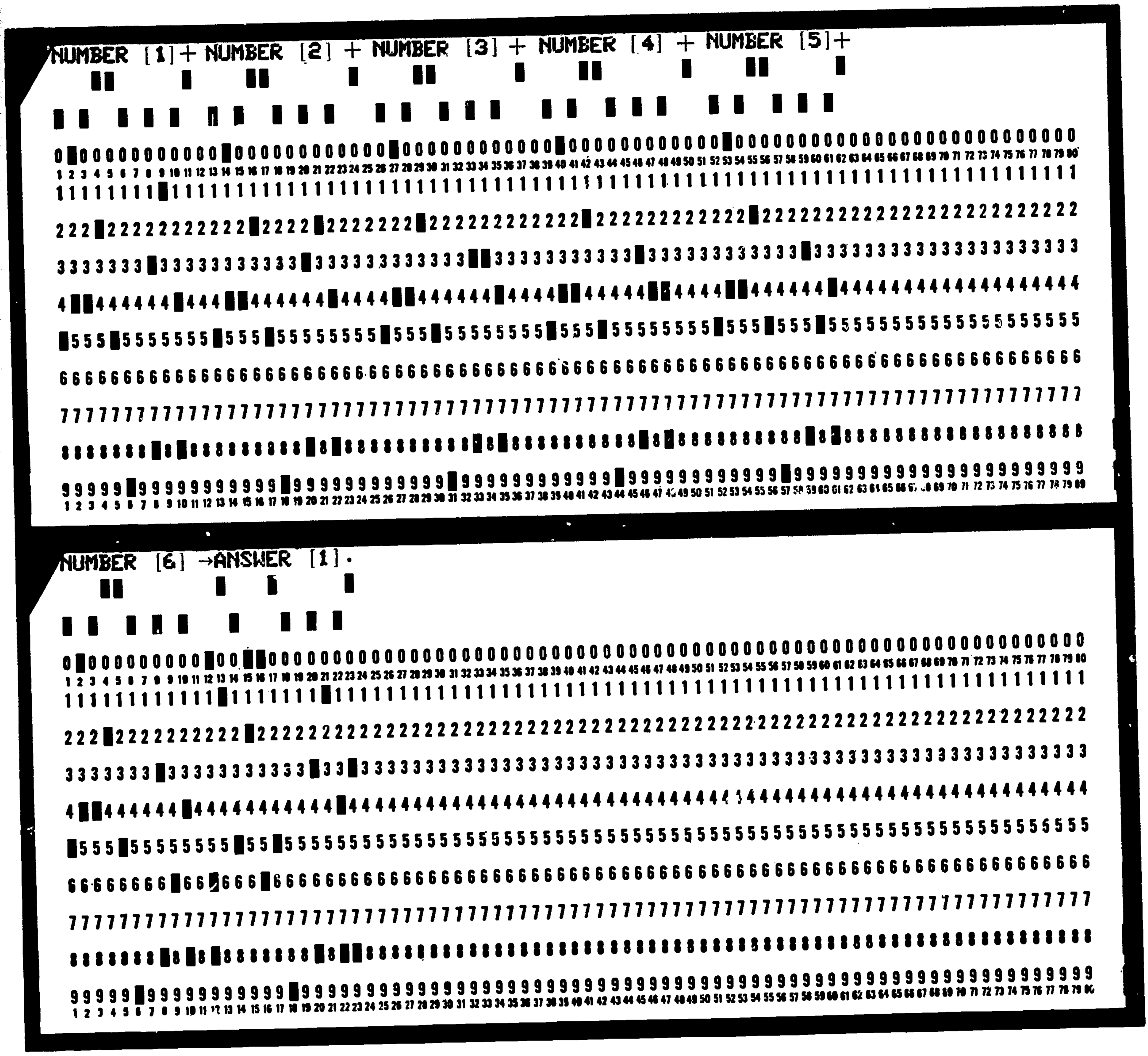


Fig. 2-14. Compiler Statement Cards

compiler to mean: "Put whatever is on the left side of the arrow into the location designated on the right side." (Notice how much this method of writing resembles our usual day-to-day writing.)

The two cards shown in Fig. 2-14 contain a statement of the problem. When they are entered into the storage unit by the card reader, as before, the compiler will interpret this set of symbols and, by *itself*, produce instructions such as those listed for the problem discussed in Part II of this chapter under "Programming for Addition."

As programs are prepared for the compiler, it will be seen that there is more than one way of writing statements for solving a problem. In this sense there is no one "right" way of writing statements. There is, of course, a "best" way, depending on whether or not factors such as the programmer's time or computer operation time need to be considered.

A Program Involving Multiplication, Addition, and Division

Consider another problem: Find the areas of four rectangles the sides of which are given. Store the value for each area in a separate position. Add all the areas and divide by 4 to obtain the average. Finally, store the average value thus obtained.

First, the names:

```
SIDE (4),
OTHER SIDE (4),
AREA (4),
AVERAGE (1),
HOLD (1),
TOTAL = 4;
```

will be entered into the computer on the card shown in Fig. 2-15.

As before, the compiler will set aside four locations to accommodate one side for each rectangle, four locations for the other side of each rectangle, four locations for the four areas, one location for the average, and one location called HOLD. (The names can be listed in any order. Their sequence is independent of their individual use in the statements which follow.) The location known as HOLD (1) (it can, of course, be called anything else) is used to "hold" the sum while preparing to obtain the average by dividing by 4. The location called TOTAL will have the number 4 stored in it.

NOTE. At this point, it may be of interest to get a glimpse of the compiler action. The compiler reads the symbols from left to right across successive cards and obtains its internal translatory instructions from the sequence of symbols it finds. For example, whenever *alphabetic* characters are found to the *right* of a *comma* and to the *left* of an *equal sign* (in this case, the word TOTAL), the location designated by these alphabetic characters has put into it the quantity found to the *right* of the *equal sign* (in this case, 4). In like manner, symbols *between commas*, such as, AREA (4), are interpreted as storage locations, *provided* no semicolon has yet been read. Symbols *between* a *comma* and a *semicolon* are interpreted as the *last* storage location in the list. These examples merely illustrate the procedure—in general, the compiler knows what to do according to rules that were originally programmed into it and which in turn are based on particular combinations of pairs of symbols as they appear in the sequence that, starting from the be-

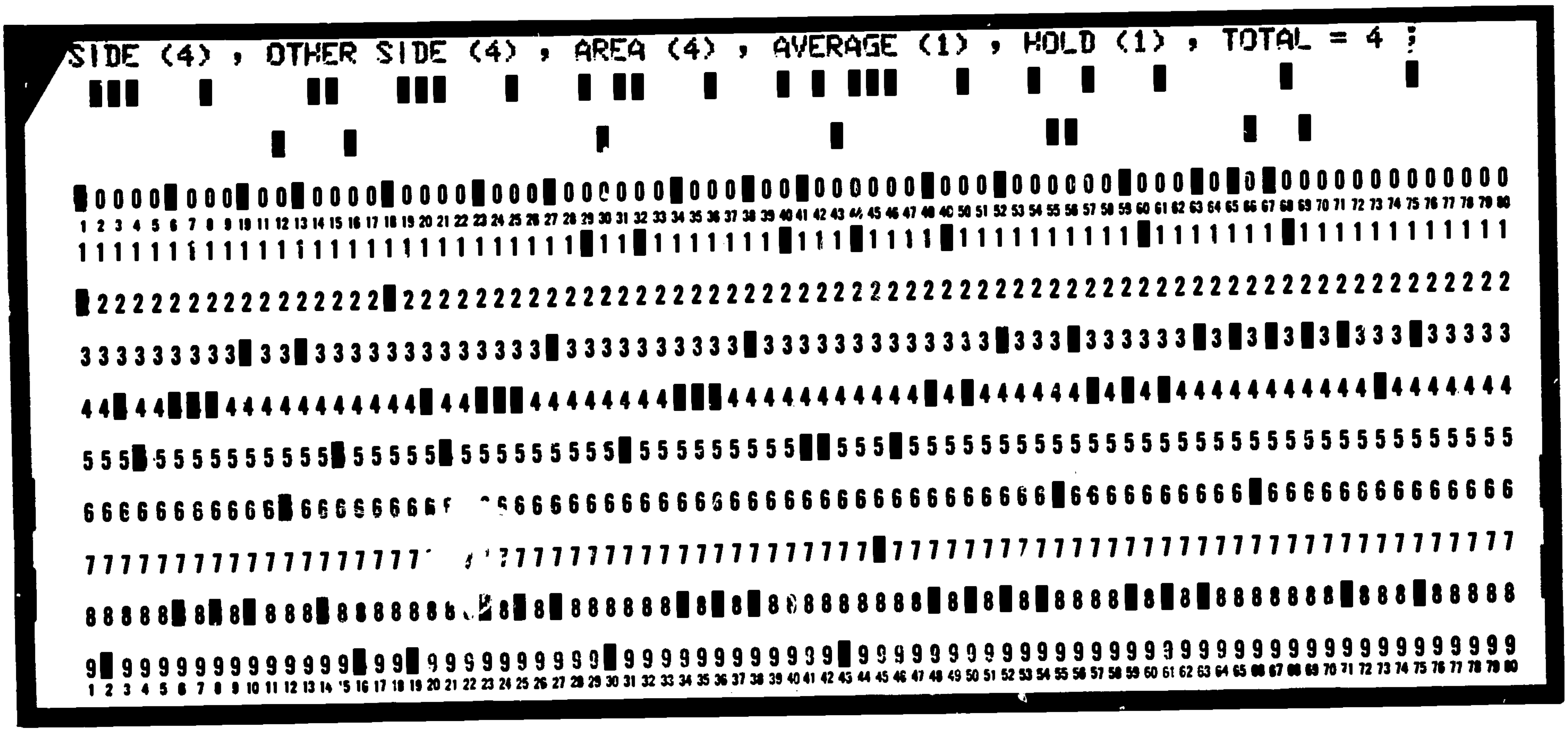


Fig. 2-15. Compiler Name Card: *Compute the areas for four rectangles and find the average.*

ginning, constitutes the program. As in any language, the punctuation helps to make the meaning clear.

Second, the statements:

The program listed below has explanatory remarks. Again, these statements would be punched on cards and entered into the computer through the card reader. Note that in writing statements square brackets [ ] are used to indicate particular values within a group specified by name.

Statements	Explanatory remarks
SIDE [1] × OTHER SIDE [1] → AREA [1],	Obtain area of first rectangle. (The symbol × means multiply.)
SIDE [2] × OTHER SIDE [2] → AREA [2],	Obtain area of second rectangle.
SIDE [3] × OTHER SIDE [3] → AREA [3],	Obtain area of third rectangle.
SIDE [4] × OTHER SIDE [4] → AREA [4],	Obtain area of fourth rectangle.
AREA [1] + AREA [2] + AREA [3] + AREA [4] → HOLD [1],	Obtain sum of the four rectangles and store sum temporarily in location called HOLD [1].
HOLD [1]/TOTAL → AVERAGE [1].	Divide the sum (HOLD [1]) by 4 (TOTAL) and put result in location called AVERAGE [1]. (The symbol / means divide.)

Consolidating Repetitive Steps

Although the above steps are simple, it is time-consuming to write them; this would be especially so if, instead of 4 rectangles, there were 400 rectangles.

The compiler has been developed so that the repetitive steps can be consolidated. This can be described by considering the idea of subscripts, so often used in mathematics. The symbol X_i indicates that, whatever is represented by X, it has several values; the i refers to the fact that a particular value of X may now be designated and that later another value of X may be designated; and so on. For example, if y equals the sum of ten consecutive values of the expression $(ax + b)$, where a and b are constants, this can be written as:

$$y = \sum_{i=1}^{i=10} (ax_i + b).$$

Here the Greek letter $\sum$, called Sigma, means "obtain the sum" and refers to the sum of the ten values of the expression $(\quad + b)$ developed by using different values of x, starting with the value referred to as No. 1 and ending with the value referred to as No. 10.

It is often the case that a procedure starts with a value and proceeds one-by-one until the last one is accounted for. A previous example was of this kind.

In the example above the "first" SIDE is multiplied by the "first" OTHER SIDE and the result is placed in the "first" AREA. Then the "second" SIDE is multiplied by the "second" OTHER SIDE and the result is placed in the "second" AREA. That is, the process is "stepping" through three separate parts of storage, as follows: (1) starting with the *first* one, (2) step *one* at a time; (3) do it *four* times. These three items can be written as follows:

1	(1)	4
↗	↑	↖
Start with this one (first)	Size of step (1)	How many values to be considered? (4)

In like manner, 1 (2) 25 means (1) start with the *first* one, (2) step *two* at a time, (3) continue until *25* values have been considered. (Even if it was skipped it was considered.) This involves the first, third, fifth, seventh, etc., up to the twenty-fifth value, or 13 values in all, describing how i as a subscript, as in X_i, is to be treated. In the statements below, i is written as the upper-case I and is sometimes called an *index*. When used in writing a program, the value of I precedes the other steps in order to tell the compiler in advance what to expect. The problem involving the rectangles is written below, using indexing.

```
Names:
    SIDE (4),
    OTHER SIDE (4),
    AREA (4),
    AVERAGE (1),
    HOLD (1),
    TOTAL = 4;
Statements:
    I = 1 (1) 4
    {SIDE [I] × OTHER SIDE [I] → AREA [I],},
         AREA [1] + AREA [2] + AREA [3] + AREA [4] → HOLD [1],
         HOLD [1]/TOTAL → AVERAGE [1] ..
```

The compiler has been developed so that the repetitive steps can be consolidated. This can be described by considering the idea of subscripts, so often used in mathematics. The symbol X_i indicates that, whatever is represented by X, it has several values; the i refers to the fact that a particular value of X may now be designated and that later another value of X may be designated; and so on. For example, if y equals the sum of ten consecutive values of the expression $(ax + b)$, where a and b are constants, this can be written as:

$$y = \sum_{i=1}^{i=10} (ax_i + b).$$

Here the Greek letter $\sum$, called Sigma, means "obtain the sum" and refers to the sum of the ten values of the expression $(ax + b)$ developed by using different values of x, starting with the value referred to as No. 1 and ending with the value referred to as No. 10.

It is often the case that a procedure starts with a value and proceeds one-by-one until the last one is accounted for. A previous example was of this kind.

In the example above the "first" SIDE is multiplied by the "first" OTHER SIDE and the result is placed in the "first" AREA. Then the "second" SIDE is multiplied by the "second" OTHER SIDE and the result is placed in the "second" AREA. That is, the process is "stepping" through three separate parts of storage, as follows: (1) starting with the *first* one, (2) step *one* at a time; (3) do it *four* times. These three items can be written as follows:

1	(1)	4
↗	↑	↖
Start with this one (first)	Size of step (1)	How many values to be considered? (4)

In like manner, 1 (2) 25 means (1) start with the *first* one, (2) step *two* at a time, (3) continue until *25* values have been considered. (Even if it was skipped it was considered.) This involves the first, third, fifth, seventh, etc., up to the twenty-fifth value, or 13 values in all, describing how i as a subscript, as in X_i, is to be treated. In the statements below, i is written as the upper-case I and is sometimes called an *index*. When used in writing a program, the value of I precedes the other steps in order to tell the compiler in advance what to expect. The problem involving the rectangles is written below, using indexing.

```
Names:
    SIDE (4),
    OTHER SIDE (4),
    AREA (4),
    AVERAGE (1),
    HOLD (1),
    TOTAL = 4;
Statements:
    I = 1 (1) 4
    {SIDE [I] × OTHER SIDE [I] → AREA [I],},
          AREA [1] + AREA [2] + AREA [3] + AREA [4] → HOLD [1],
          HOLD [1]/TOTAL → AVERAGE [1] ..
```

The names are the same as before. There are two differences in the calculation of the areas. First, the cyclic nature of the procedure is specified by $I = 1\ (1)\ 4$. Second, the part of the process which is to be repeated according to I is enclosed in braces, { }. These are simply symbols which the compiler interprets as follows: "The statement inside the braces is going to be repeated in a manner specified by the value of I." Parentheses, if employed within the braces, are used in the usual manner to enclose an operation.

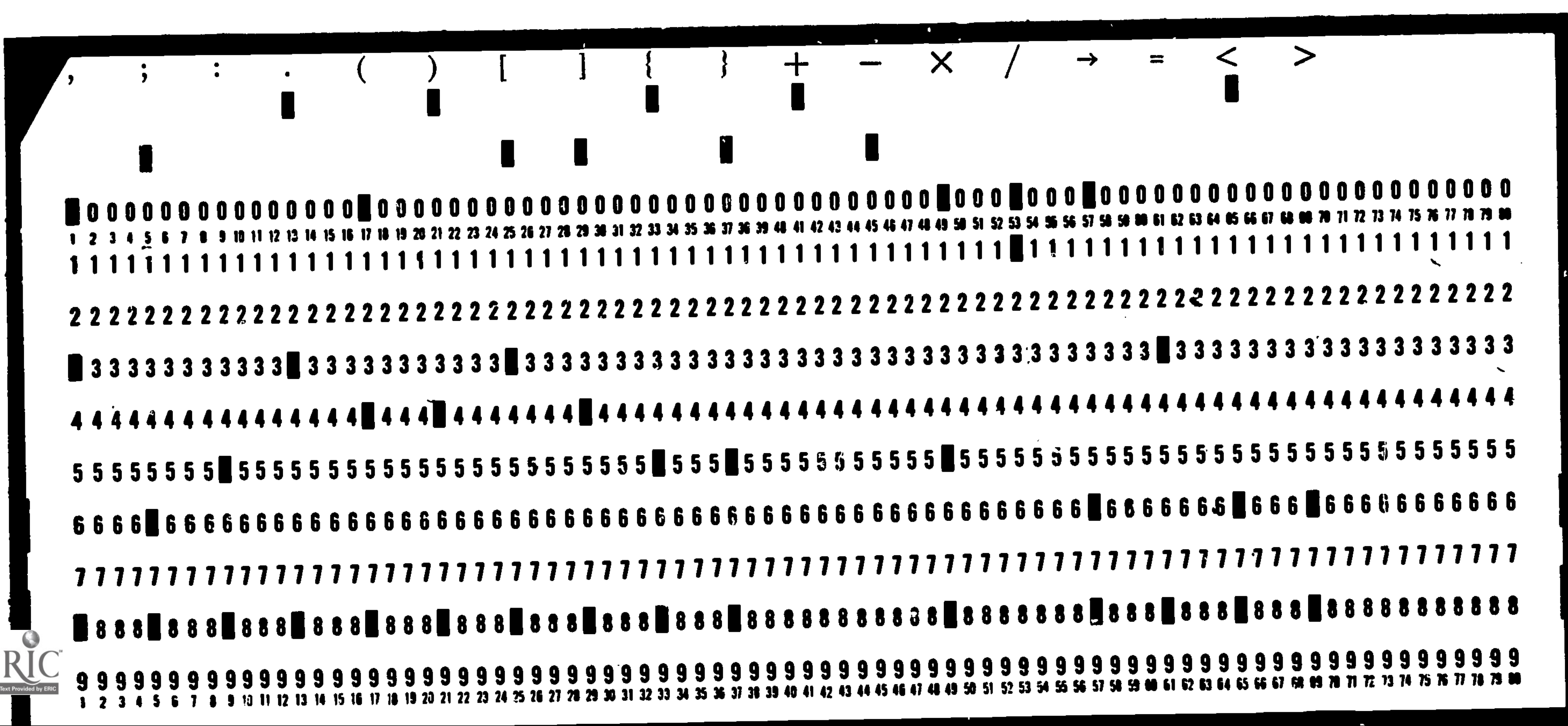
The symbols corresponding to {, }, [,], and others, are not available on the standard keypunch keyboard, but can be provided just as the symbols (,), and / are provided. In some computer laboratories such symbols are used, and for this discussion the COMPIAC installation is assumed to have this capability.

The card in Fig. 2-16 illustrates the kinds of codes which could be used to indicate these symbols. It should be pointed out that the symbol for multiply, $\times$, is not the same as the letter x in the alphabet.

Still another refinement is possible by considering, again, the symbol $\rightarrow$. This means, as stated above: "Put whatever is on the left side of the arrow in the location on the right side." Suppose the statement is as follows:

ANSWER [1] + RESULT [1] → RESULT [1].

This would be interpreted by the compiler, on a given cycle of a repetitive sequence, as follows: "Add the quantity in the location called ANSWER [1] to the quantity in the location called RESULT [1] and put this sum back into the location called RESULT [1]." In the example it would have been possible



to accumulate, at each step, the separate areas in location HOLD [1], rather than to put each area in its own location and then finally to accumulate in HOLD [1]. This would be written as follows:

```
I = 1 (1) 4
{HOLD [1] + SIDE [I] × OTHER SIDE [I] → HOLD [1],},
```

In other words, whatever is in HOLD [1] is added to the product of SIDE [*I*] × OTHER SIDE [*I*] and, according to the arrow, put back in HOLD [1] again. Now, of course, HOLD [1] contains what it originally held, plus the product.

This loop is continued according to the value of *I*, as before. Of course, to start this loop it is necessary to be sure that HOLD [1] is zero at the beginning—that is, cleared of other numbers and ready to accept the accumulation. This can be accomplished by simply putting zero into HOLD [1] to start with, as shown below.

```
Names:
  SIDE (4),
  OTHER SIDE (4),
  AVERAGE (1),
  HOLD (1),
  TOTAL = 4;
Statements:
  0 → HOLD [1],
  I = 1 (1) 4
  {HOLD [1] + SIDE [I] × OTHER SIDE [I] → HOLD [1],},
  HOLD [1]/TOTAL → AVERAGE [1]..
```

NOTE. AREA (4) is no longer needed, since accumulations will be performed in HOLD.

Computer Punctuation

Before developing additional parts of the compiler language a point should be emphasized. The language discussed consists of statements separated by punctuation marks in a way similar to the statements in English. The rules of compiler language are more rigid, however, than those of English; in compiler language only certain types of statements are acceptable, and the proper use of punctuation is necessary, as shown below.

The *comma* is used to separate (1) one name from the next and (2) one program statement from the next.

The *semicolon* is used to signify the end of the names.

The *colon* has two uses:

1. To separate a group of words, as, for example:

```
FIRST PART OF ANSWER:
  I = 1 (1) 4
  {SIDE [I] × OTHER SIDE [I] → AREA [1],},
```

This means that the programmer has chosen to separate the problem into parts; thus, the statements following the colon are designed to produce what he has chosen to call FIRST PART OF ANSWER. The symbols or words preceding the colon constitute a title or label that identifies the group of statements appearing after the colon.

2. The colon is used in making a comparison and selecting alternatives accordingly. For example, a problem may take one of two directions, depending on the relative size of two numbers, A and B, such that if A is larger, a calculation called REAL ROOTS is performed; otherwise, if B is larger, a calculation called IMAGINARY ROOTS is performed. This is written as follows and illustrates the second use of the colon:

```
A > B: REAL ROOTS. IMAGINARY ROOTS.
       IMAGINARY ROOTS:
       STATEMENT,
       STATEMENT,
       STATEMENT.

       REAL ROOTS:
       STATEMENT,
       STATEMENT,
       STATEMENT.
```

The compiler will interpret the above as follows: "If the value A is greater than the value B, go to the part of the problem entitled REAL ROOTS and continue from that point. If A is not larger than B, go to the part of the problem entitled IMAGINARY ROOTS and continue from that point."

The comparison statements always take the same form—the "True" alternative is immediately after the comparison, the "False" alternative is in the second position after the comparison. It does not matter where the two other parts of the program are located: the compiler will locate the proper part and continue.

In addition to the "greater than" symbol, $>$, comparison can be based on the "less than" symbol, $<$, and the "equal" symbol, $=$.

The *period* is used in two ways:

1. The *single period* signifies that a part of the program has ended and the next part may be out of the usual sequence. For example, if A were greater than B, the REAL ROOTS part would have been done, but the period after REAL ROOTS indicates that the next part to be done is out of sequence; that is, the IMAGINARY ROOTS part is skipped.

2. A *double period* means that the end of the program has been reached.

The explanation of these punctuation rules shows that they are not much different from the usual use of the punctuation marks in writing ordinary English.

The following example demonstrates the use of punctuation rules: There are 100 pairs of numbers, corresponding to 100 figures, which include both squares and rectangles. The first number of a given pair is one side of the figure, and the second number of a given pair is the other side of the figure. Assume that the numbers are presented in the following order, and that there is at least one of each type of figure.

1	First number of first figure
2	Second number of first figure
3	First number of second figure
4	Second number of second figure
5	First number of third figure
6	Second number of third figure
etc.	etc.

PROBLEM. Compute the perimeter of all figures; also, compute the area if the figure is a square. Determine the average area for the figures that are squares.

The problem can be solved in more than one way. The particular program shown below lists the necessary names followed by the parts of the problem. There are seven distinct parts of this problem and they have the following titles.

1. SET CERTAIN VALUES
2. START OF PROBLEM
3. CALCULATE FOR SQUARES
4. CALCULATE FOR RECTANGLES
5. RETURN
6. GET AVERAGE FOR SQUARES
7. STOP

Other words can be used for the titles. Each of the seven is marked by a colon. Notice how the seven parts of the problem can be made to fit a flow diagram (see Fig. 2-17). Also notice that it is possible to choose one's own words and to arrange the operations in much the same way as when a problem is written on the blackboard or explained to another person.

Names	Explanatory notes
NUMBER (200), RECTANGLE PERIMETERS (100), SQUARE PERIMETERS (100), SQUARE AREAS (100), AVERAGE FOR SQUARES (1), TOTAL SQUARES (1), SUM AREAS (1);	This is the list of names, ending with a semicolon. The names are of the same type as mentioned earlier. Some of the values of 100 will probably not be used, but they represent the maximum number possible. That is, it is not known in advance how many squares there will be, but there will not be more than 100; thus, if 100 is used in the name list, it will be adequate. TOTAL SQUARES is the name of a location that will be reset to zero in preparing to count the squares. (See statement below.)

Statements	Explanatory notes
SET CERTAIN VALUES: $1 \rightarrow K \rightarrow L$, $0 \rightarrow$ TOTAL SQUARES [1],	$1 \rightarrow K \rightarrow L$ means that K is set to equal 1, and then L is set to equal K. As a result, K and L both are equal to 1. K will be used to keep track of squares. L will be used to keep track of rectangles.
START OF PROBLEM: $I = 1$ (2) 199 {NUMBER [I] − NUMBER [$I + 1$] = 0: CALCULATE FOR SQUARES. CALCULATE FOR RECTANGLES.	The expression $I = 1$ (2) 199 is similar to that already described. It means that whenever I is referred to, a sequence of operations follows which, in this case, starts with the first one and proceeds by steps of 2 until 199 is reached. The first I corresponds to "the first number of the first figure". The $I + 1$ corresponds to "the second number of the first figure." I and $I + 1$ refer to pairs of numbers in the original list of 100 pairs. The braces, { }, enclose the statements to be performed repetitively under the control of I. Note that the closing brace is not reached until we arrive at RETURN:} (see below). The first statement within the braces is for determining whether the figure is a square, i.e., whether one

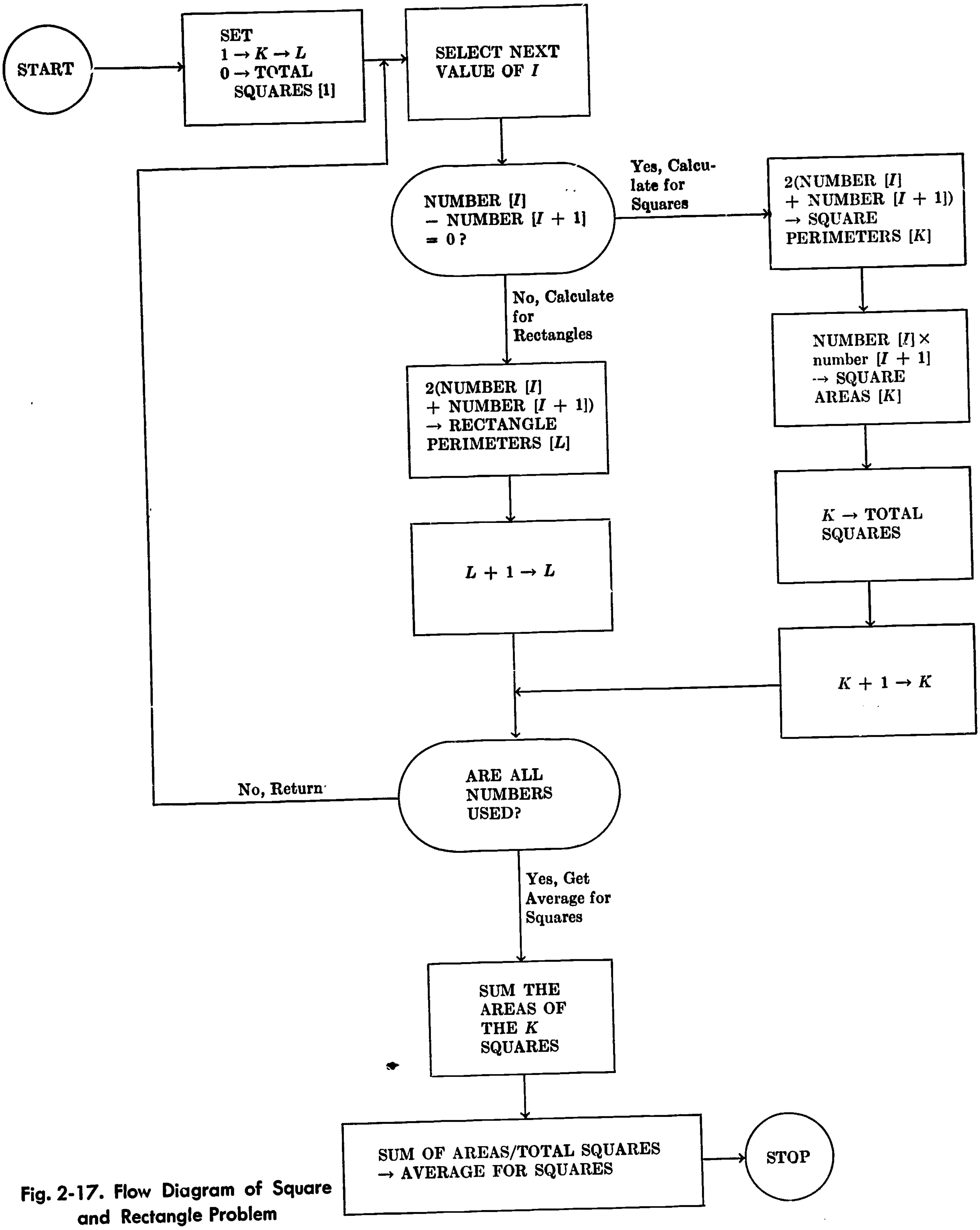


Fig. 2-17. Flow Diagram of Square and Rectangle Problem

Statements | Explanatory notes

side subtracted from the other side equals zero. If the difference is zero, the figure is a square, and the next operation is the part CALCULATE FOR SQUARES.

If the difference is not zero, the figure is a rectangle; and the next operation is the part CALCULATE FOR RECTANGLES.

```
CALCULATE FOR SQUARES:
  2 × (NUMBER [I] +
    NUMBER [I + 1])
  → SQUARE PERIMETERS
    [K],
  NUMBER [I] × NUMBER
    [I + 1] →
  SQUARE AREAS [K],
  K → TOTAL SQUARES [1]
  K + 1 → K,
  RETURN.
```

The first statement obtains a perimeter for the first figure and puts the result in the first (since K is now equal to 1) location of the storage part reserved under the name SQUARE PERIMETERS. The same applies to the area put into the first location of what was reserved under the name SQUARE AREAS. (K is still 1 at this stage and is later increased to 2, for the next time around.)

The remainder of the CALCULATE FOR SQUARES part first records the value of K in the location called TOTAL SQUARES. This will be needed to obtain the average later on.

```
CALCULATE FOR
    RECTANGLES:
  2 × (NUMBER [I] +
    NUMBER [I + 1])
  → RECTANGLE
    PERIMETERS [L],
  L + 1 → L,
  RETURN.
  RETURN:},
```

K is now increased by 1.

The last statement "sends" control to the place called RETURN:, and then, because of the brace, }, to the beginning of the cycle.

At this stage the value of I is adjusted by 2.

Notice that, regardless of the alternative, i.e., square or rectangle, the last operation is to "send" the control to RETURN:. Thus, in either case, the direction to return to the beginning of the cycle originates from the same place and is signalled by the brace, }.

K and L are used in somewhat the same way as I, except that it is not known in advance how many K's or L's there will be. (Initially K and L were both set equal to 1.) After each time that K or L are referred to they will be increased by 1 (by the procedure $K + 1 \rightarrow K$, instead of by the method used for I).

By the time this cycle has been repeated until I has reached the value 199, a certain number (K) of perimeters and areas of squares will have been stored; also a certain number (L) of perimeters of rectangles will have been stored. When I reaches its limit, control will go to the place to the right of the brace, }. A similar explanation would apply to the CALCULATE FOR RECTANGLES part.

```
GET AVERAGE FOR
    SQUARES:
  0 → SUM AREAS [1],
  M = 1 (1) TOTAL SQUARES
  [1] {SQUARE AREAS [M] +
    SUM AREAS [1]
  → SUM AREAS [1],},
  SUM AREAS [1]/TOTAL
    SQUARES [1]
  → AVERAGE FOR
    SQUARES [1],
  STOP.
```

In the part GET AVERAGE FOR SQUARES the first action is to set the location called SUM AREAS to zero. This is necessary prior to accumulating the areas of the squares. As before, M is a value which, when referred to in the repetitive operation to follow, will start with the first one and increase by steps of one, up to and including TOTAL SQUARES. But TOTAL SQUARES has already been set to the value of K, which is the number of squares.

The statement in the braces, { }, accumulates the areas of squares. This is done by adding the first area ($M = 1$ initially) to SUM AREAS (initially reset to

Statement	Explanatory notes
	zero) and putting the total back in SUM AREAS. For the next cycle, $M = 2$, and this second area is added to the first, and the sum put back where the first sum had been, and so forth, until it has been done as many times as TOTAL SQUARES.
STOP:..	The double period indicates the end of the program. NOTE. This program can also be terminated as follows: AVERAGE FOR SQUARES [2], ...

Fig. 2-18 shows how some of the compiler instructions used in the above example may be written on a standard form. From this the key-punch operator punches the cards, which are then read into the computer with the data cards needed in the computation.

The following example shows the program for obtaining the real roots for a group of ten quadratic equations which have the form:

$$ax^2 + bx + c = 0$$

and in which all b's and c's are not equal to zero.

```
  A (10),
  B (10),
  C (10),
  DISCRIMINANT (1),
  RESULT = −0,
  FIRST ROOT (10),
  SECOND ROOT (10),
  SQUARE ROOT (1);
DETERMINE ROOTS:
  I = 1 (1) 10
  {B[I] × B[I] − 4 × (A[I] × C[I]) → DISCRIMINANT [1],
  DISCRIMINANT [1] < 0: NOT REAL. REAL.
NOT REAL:
  RESULT → FIRST ROOT [I] → SECOND ROOT [I],
  RETURN.
REAL:
  SQRT (DISCRIMINANT [1]) → SQUARE ROOT [1],
  (−B [I] + SQUARE ROOT [1])/(2 × A [I]) → FIRST ROOT [I],
  C [I]/(A [I] × FIRST ROOT [I]) → SECOND ROOT [I],
  RETURN.
RETURN:},
STOP.
STOP:..
```

NOTE 1. Although not previously defined, the compiler can be programmed so that symbols such as SQRT, like SIN, COS, etc., will perform the proper operations upon the quantity in parentheses immediately following, and locate the result as directed.

NOTE 2. The compiler follows strict rules of order of arithmetic operations in that multiplications and divisions take precedence over additions and subtractions, and that operations of the same precedence level are performed from left to right. For example, $A + B/C$, without a strict rule, could be interpreted as $A + (B/C)$ or as $(A + B)/C$. Since division takes precedence over addition, $A + B/C$ would be interpreted by the

Instructions	Comments
NUMBERS (200),	
RECTANGLE PERIMETERS (100),	
SQUARE PERIMETERS (100),	
SQUARE AREAS (100),	State names
AVERAGE FOR SQUARES (1),	
TOTAL SQUARES (1),	
SUM AREAS (1);	
SET CERTAIN VALUES:	Initial settings K = L = 1
1 → K → L, 0 → TOTAL SQUARES [1],	
START OF PROBLEM:	
I = 1(2)199	
{NUMBER[I] − NUMBER[I+1] = 0: CALCULATE	If $s_1 - s_2 = 0$, a square.
FOR SQUARES. CALCULATE FOR RECTANGLES.	If $s_1 - s_2 \neq 0$, a rectangle
CALCULATE FOR SQUARES:	
2 ⊗ (NUMBER[I] + NUMBER[I+1]) → SQUARE	$2(s_1 + s_2) = P$
PERIMETERS[K], NUMBER[I] ⊗ NUMBER[I+1] →	$s_1 \times s_2 = A$
SQUARE AREAS[K],	
K → TOTAL SQUARES, K+1 → K, RETURN.	Add 1 to total squares, do next pair
CALCULATE FOR RECTANGLES:	

Fig. 2-18. COMPIAC Compiler Coding Sheet

compiler as $A + (B/C)$. In the case of the same precedence level, such as $A/B \times C$, the compiler, going from left to right, would make this interpretation: $(A/B)C$. Of course, parentheses can be used to eliminate ambiguity as in the REAL part of the above program.

This procedure sets the starting value, step size and range of I. Then, within the braces, the value of the discriminant is obtained. If it is less than zero, the roots are not real, and the number -0 is put in the locations where the real roots of that particular equation would have been placed. This value, -0, merely indicates that the result was imaginary. It is, thus, distinguishable from any real roots that might be computed. If the roots are real, the first one may be obtained from the quadratic formula and the second one may be obtained by dividing the known term C by the product of the first root and the coefficient of X^2. That is, if the roots are X_1 and X_2,

$$X_1 \times X_2 = \left(\frac{-B + \sqrt{B^2 - 4AC}}{2A}\right)\left(\frac{-B - \sqrt{B^2 - 4AC}}{2A}\right) = \frac{C}{A}$$

and

$$X_2 = C/AX_1.$$

The procedures for Input and Output have not been described for the compiler language. For this discussion they are considered sufficiently similar to those already mentioned in the part on machine language that they will not be described.

References

The reader who is interested in further detail on the subject of compiler and machine language programming should consult the references listed below, which are identified in the Bibliography, Appendix C: 24, 29, 35c, 35f, 35i, 39, O3, P2.

EXERCISES

1. Write COMPIAC compiler language instructions for solving the problems described in Exercise 5, at the end of Part II of this chapter. In part 5 (a) use the following names: INTEGERS (100) for data, and COUNT (1) for the location of the number of nonzero elements. In part 5 (b), name the data NUMBERS (50), and name the location for the greatest value GREATEST (1).

HINT. Consider beginning and comparing the "next" number with its successors and if the "next" number is greater than some successor, exchange the pair.

2. (a) Write a set of compiler instructions that will direct COMPIAC to sum the squares of the first 100 integers, that is:

$1^2 + 2^2 + 3^2 + \ldots + 100^2$ (b) Write a compiler program that will set up a loop to sum the first 50 factorials, that is:

$$1 + 1\cdot 2 + 1\cdot 2\cdot 3 + \ldots + 1\cdot 2\cdot 3\cdot \ldots \cdot 49\cdot 50.$$

3. Have COMPIAC prepare a table of values of x and y for $y = 2^x$ using integral values of x such that $y < 10{,}000$.

4. (a) Sum the first n terms of the arithmetic progression $a, a + d, a + 2d, \ldots a + (n - 1)d$ using arbitrarily selected values of a, d, and n. For this exercise, do not use the formula $S_n = \frac{n}{2}(a + L)$ where L stands for the last term in the series. (b) Sum the first n terms of the geometric progression $a, ar, ar^2, \ldots ar^{n-1}$ using arbitrarily selected values of a, r, and n. Do not use the formula:

$$S_n = \frac{a(r^n - 1)}{(r - 1)}.$$

5. Many of the series developed in calculus provide interesting material. Have COMPIAC approximate the value of the following functions correct to the nearest ten-thousandth.

$$\text{(a)} \quad e = 1 + \frac{1}{1!} + \frac{1}{2!} + \frac{1}{3!} + \frac{1}{4!} \cdots$$

$$\text{(b)} \quad \sin x = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \cdots$$

$$\text{(c)} \quad e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!} + \cdots$$

6. Write COMPIAC compiler programs for Exercises 6, 9, 15, 18, 19, and 20 at the end of Chapter 1.

Chapter 3

REPETITIVE PROCESSES AND APPROXIMATIONS

Part I. Repetitive Processes

It was pointed out in the Introduction that a digital computer may be very useful in solving problems, provided that the means of solution have been properly organized and the procedures thus developed can take advantage of the computer's ability to perform operations rapidly and repetitively. This chapter discusses some aspects of repetitive processes and shows how they can be used in solving mathematical problems. More advanced examples of repetitive processes are given in Chapter 4.

NOTE. No attempt has been made in this book to prove the validity of the equations used. If the reader desires rigorous proofs, he should consult the references listed on page 69. This also applies to parts of Chapter 4 and the Appendix.

Example 1. Multiplication by Repeated Addition

One way to multiply one nonzero number by another is to add successively the multiplicand and subtract 1 from the multiplier for each addition. When the result of the subtraction sequence reaches zero the process is finished, with the product being equal to the final sum. As an example, use this method to multiply 12 by 5.

Step number	Multiplicand = 12 Sum	Multiplier = Difference
0	$= 00$	
1	$0 + 12 = 12$	$5 - 1 = 4$
2	$12 + 12 = 24$	$4 - 1 = 3$
3	$24 + 12 = 36$	$3 - 1 = 2$
4	$36 + 12 = 48$	$2 - 1 = 1$
5	$48 + 12 = 60$	$1 - 1 = 0$

An advantage of this method is that no multiplication table need be learned. A disadvantage is that the process is slow. For example, in order to multiply 144 by 265, either 144 additions of 265 or 265 additions of 144 are needed, depending on which number is selected as the multiplier. Figure 3-1 shows a flow diagram for multiplying by the repeated addition process.

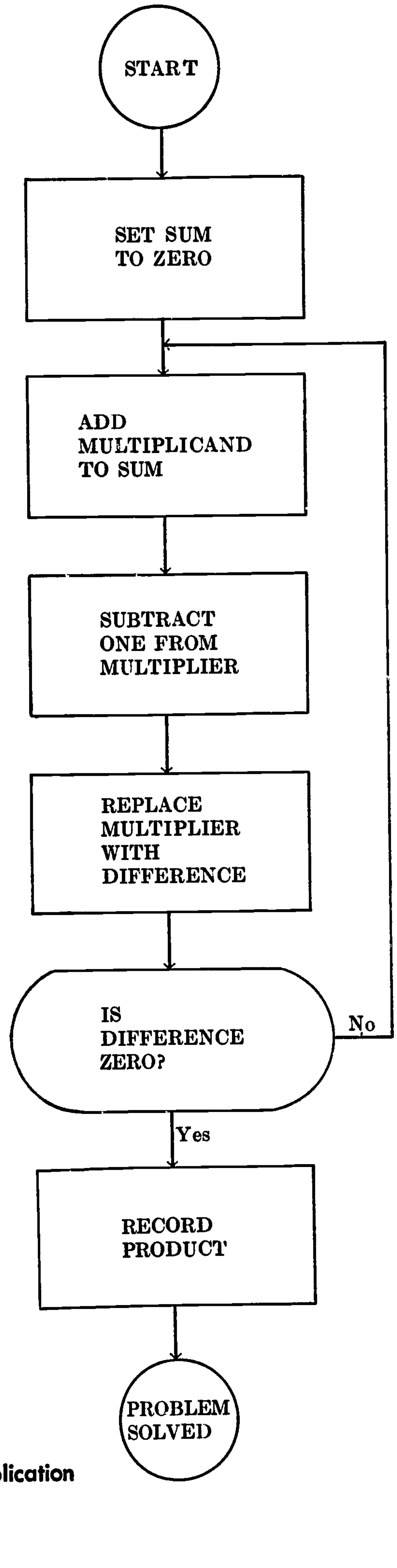


Fig. 3-1. Addition Process for Multiplication

Example 2. Multiplication by Repeated Addition and Shifting

A faster method for multiplying is the repeated addition and shifting process. It also does not require knowledge of the multiplication table. In using this method, begin, as in Example 1, by subtracting 1 from the multiplier and adding the multiplicand. However, when the right-hand digit of the remainder becomes zero, start subtracting 10 from the multiplier and adding 10 times the multiplicand to the sum. After the tens' digit of the multiplier becomes zero, begin subtracting 100 from the multiplier and adding 100 times the multiplicand. Continue this process until all the digits of the multiplier have been reduced to zero. For example, to multiply 324 by 123 one would proceed as follows:

Step number	Sum	Difference
0	000	123
1	+324	−1
	324	122
2	+324	−1
	648	121
3	+324	−1
	972	120
4	+324*	−1*
	4212	110
5	+324*	−1*
	7452	100
6	+324**	−1**
	39852	000

In Steps 4–6 the asterisks indicate the multiplications by 10 referred to above. Notice that the same effect as multiplying by 10 can be obtained by *shifting* the digits one place to the left. The dotted line segments at the end of Steps 3 and 5 indicate where shifts are to be made. The above method requires only 6 additions as compared to the 123 additions that would be necessary if the previous method were used. Fig. 3-2 shows a flow diagram for the repeated addition and shifting method.

On page 63 is a comparison of the steps needed to multiply 23 by 12 by the two processes described above as well as the ordinary method for multiplying (Method III).

Notice that the example showing the ordinary method for multiplying (Method III) is not written in the usual manner. However, if a flow diagram (as previously shown for Methods I and II) were made for this method, it would be necessary to follow a procedure similar to the one shown above. The "00" may be thought of as representing a clean sheet of paper; the first addition, as writing down the multiplicand.

In terms of speed, Method II is superior to Method I; in turn, Method III is

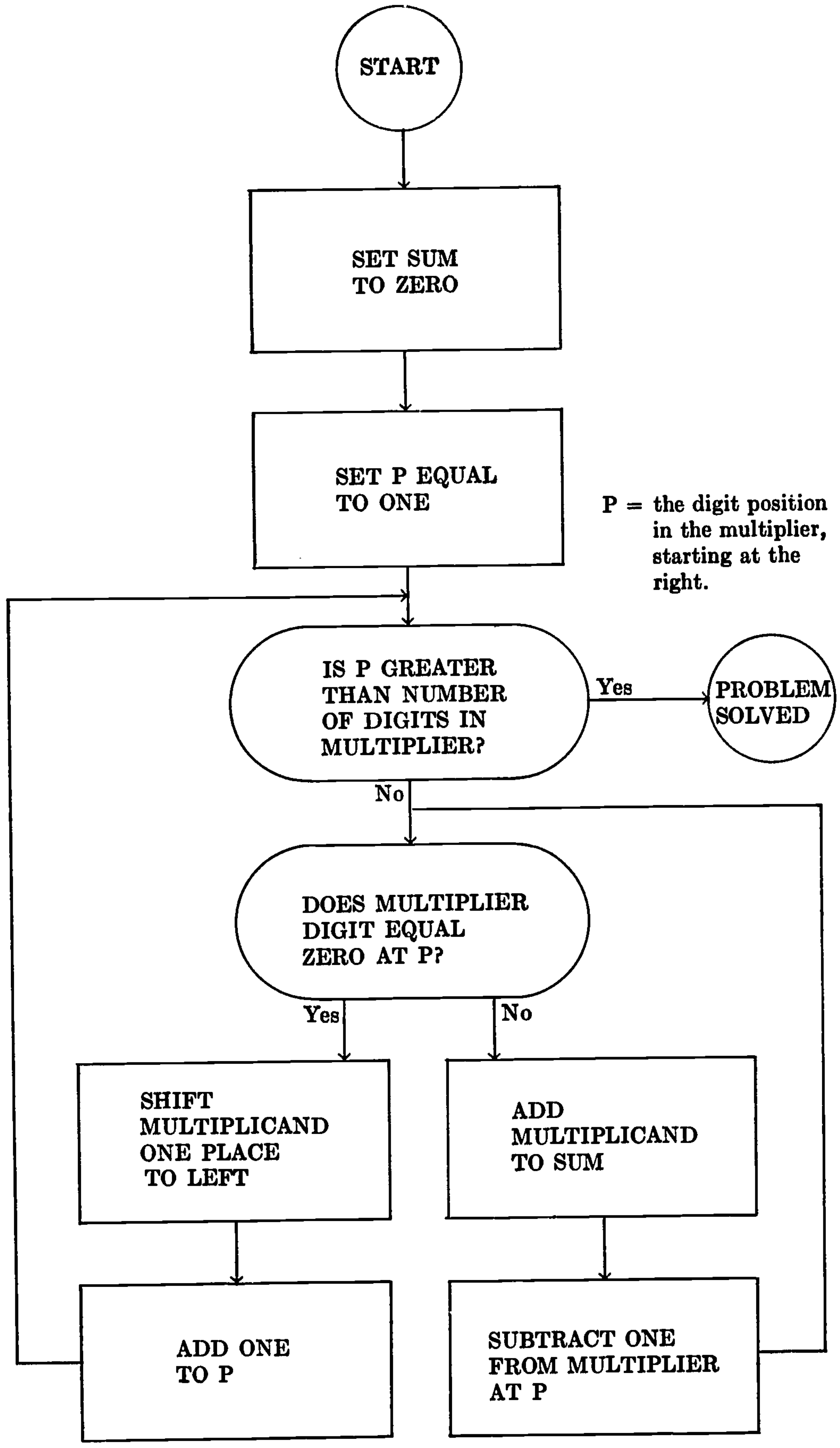


Fig. 3-2. Addition Process for Multiplication Using Digit Positions

Method I		Method II		Method III
00	12	00	12	00
+ 23	− 1	+ 23	− 1	+ 23
23	11	23	11	23
+ 23	− 1	+ 23	− 1	× 2
46	10	46	10	46
+ 23	− 1	+23*	−1*	+23* (10 × 23)
69	9	276	00	276
+ 23	− 1			
92	8			
+ 23	− 1			
115	7			
+ 23	− 1			
138	6			
+ 23	− 1			
161	5			
+ 23	− 1			
184	4			
+ 23	− 1			
207	3			
+ 23	− 1			
230	2			
+ 23	− 1			
253	1			
+ 23	− 1			
276	0			

superior to Method II. However, each succeeding method requires more complex logical operations. For example, the flow diagram for Method II is more complex than the one for Method I. The number of additions in Method I is equal to the *value* of the multiplier (12). In Method II, the number of additions is equal to the *sum of the digits* of the multiplier ($1 + 2 = 3$). In Method III the number of additions is equal to the *number of digits* in the multiplier (2).

It is worth noting that Method II, repeated addition and shifting, can be applied to make a useful multiplying device out of any desk adding machine equipped with a repeat key. Since most desk adding machines can also subtract, they can be made to divide, by the procedure discussed below.

Example 3. Division by Repeated Subtraction

It is possible to divide one number by another by successively subtracting the divisor from the dividend and counting the number of subtractions necessary to reduce the remainder to a number smaller than the divisor. For example, to divide 24 by 6 proceed as follows:

	Number of subtractions	Is remainder smaller than divisor?
24		
− 6	1	
18		No
− 6	2	
12		No
− 6	3	
6		No
− 6	4	
0		Yes

Thus, in this case, $24 \div 6 = 4$, with no remainder.

To divide 27 by 5 do the following:

	Number of subtractions	Is remainder smaller than divisor?
27		
− 5	1	
22		No
− 5	2	
17		No
− 5	3	
12		No
− 5	4	
7		No
− 5	5	
2		Yes

Thus $27 \div 5 = 5$, with a remainder of 2.

Fig. 3-3 shows a flow diagram for division by the repeated subtraction process.

A question to ask at this point might be, What happens if zero is used as a divisor? The answer is, of course, that the process as diagrammed will continue going through the boxes labeled SUBTRACT DIVISOR FROM DIVIDEND, IS DIVISOR LARGER THAN DIFFERENCE?, and ADD 1 TO NUMBER OF SUBTRACTIONS without stopping, since the answer to the last question will never be *Yes* (see Fig. 3-3). This demonstrates the reason for excluding division by zero from our arithmetic. It also points out a difficulty in analyzing problems to be solved by computers: since computers follow a sequence of steps without deviating, it is the responsibility of the programmer or problem analyst

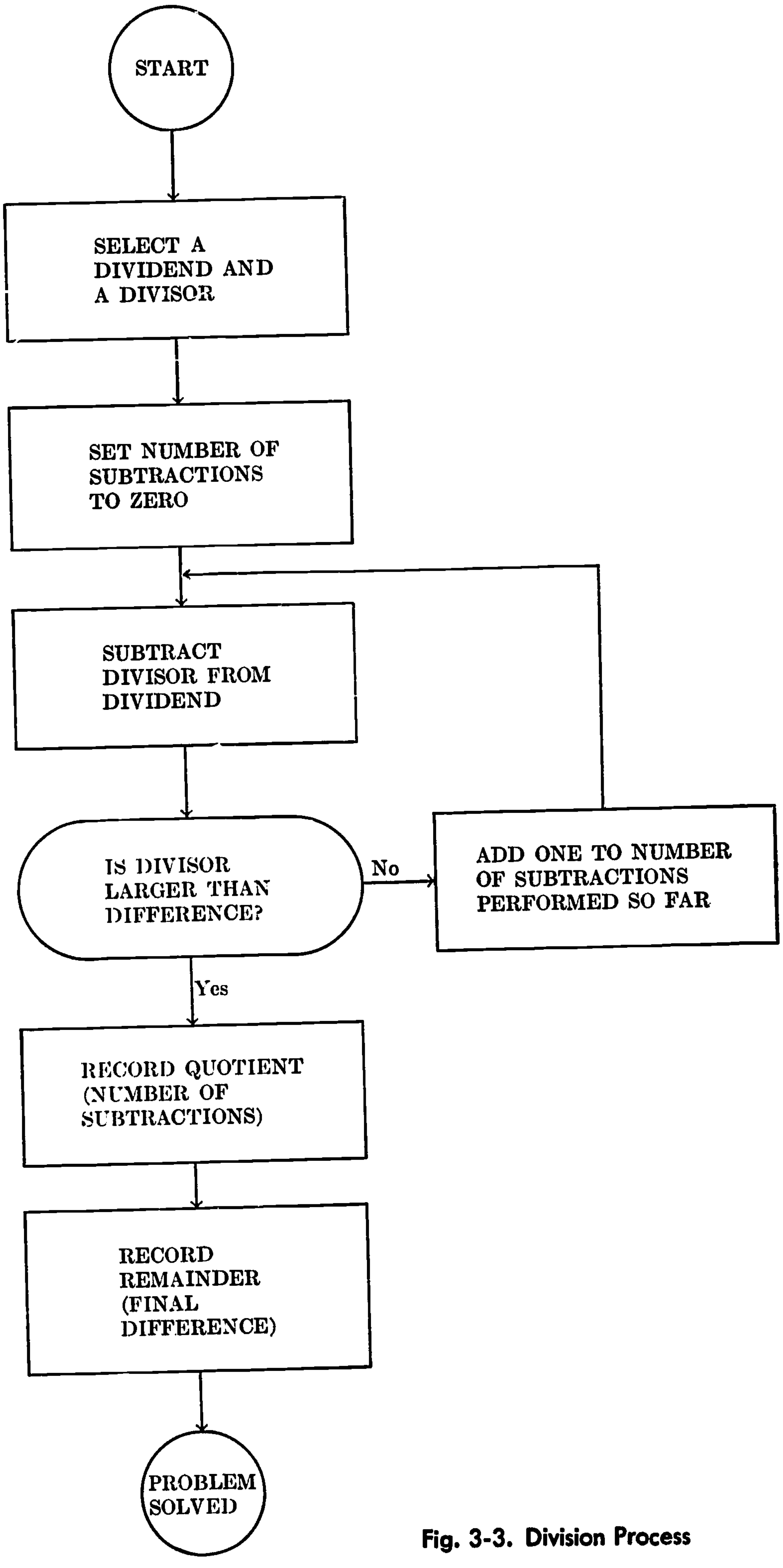


Fig. 3-3. Division Process

to ascertain that no elements exist which might lead to an unending repetition of the same set of steps. This kind of nonstop repetition is sometimes referred to as being *caught in a loop*, and care must be exercised to prevent its occurrence.

Example 4. Finding All the Prime Factors of an Integer

Suppose it is desired to find the prime factors of 210. Assume also that a list of prime numbers is available. The process, or algorithm, to accomplish this is as follows:

NOTE. For a given integer, the integer itself and 1 are not considered factors.

1. Try the first (next) prime, 2, as a factor.
   It is a factor ($210 \div 2 = 105$).
2. Try 2 again as a factor.
   It is not a factor ($105 \div 2 = 52 + 1$ remainder).
3. Try the next prime, 3.
   It is a factor ($105 \div 3 = 35$).
4. Try 3 again as a factor.
   It is not a factor ($35 \div 3 = 11 + 2$ remainder).
5. Try the next prime, 5.
   It is a factor ($35 \div 5 = 7$).
6. Try 5 again as a factor.
   It is not a factor ($7 \div 5 = 1 + 2$ remainder).
7. Try the next prime, 7.
   It is a factor ($7 \div 7 = 1$).

Since the quotient is 1, there are no more prime factors, and the process is completed. This process can be diagrammed as follows:

$$\begin{array}{r|l} 2 & \underline{210} \\ 3 & \underline{105} \\ 5 & \underline{35} \\ 7 & \underline{7} \\ & 1 \end{array}$$

Therefore, $210 = 2\cdot3\cdot5\cdot7$.

Computers can be programmed to generate prime numbers, but for the purposes of this illustration, it is assumed that a table of prime numbers, to which the computer can refer, is in storage.

A flow diagram for most of the procedure might be as shown in Fig. 3-4.

NOTE. The *next* prime can be the *first* prime as far as the computer is concerned when it begins the program. Assume that this selection process excludes the integer itself if it is also the *next* prime. After that it will change from 2 to 3 to 5, etc., on each loop.

Example 5. Finding the Greatest Common Divisor of Two Integers

Euclid, the famous Greek mathematician, is responsible for this algorithm. It involves repeated division:

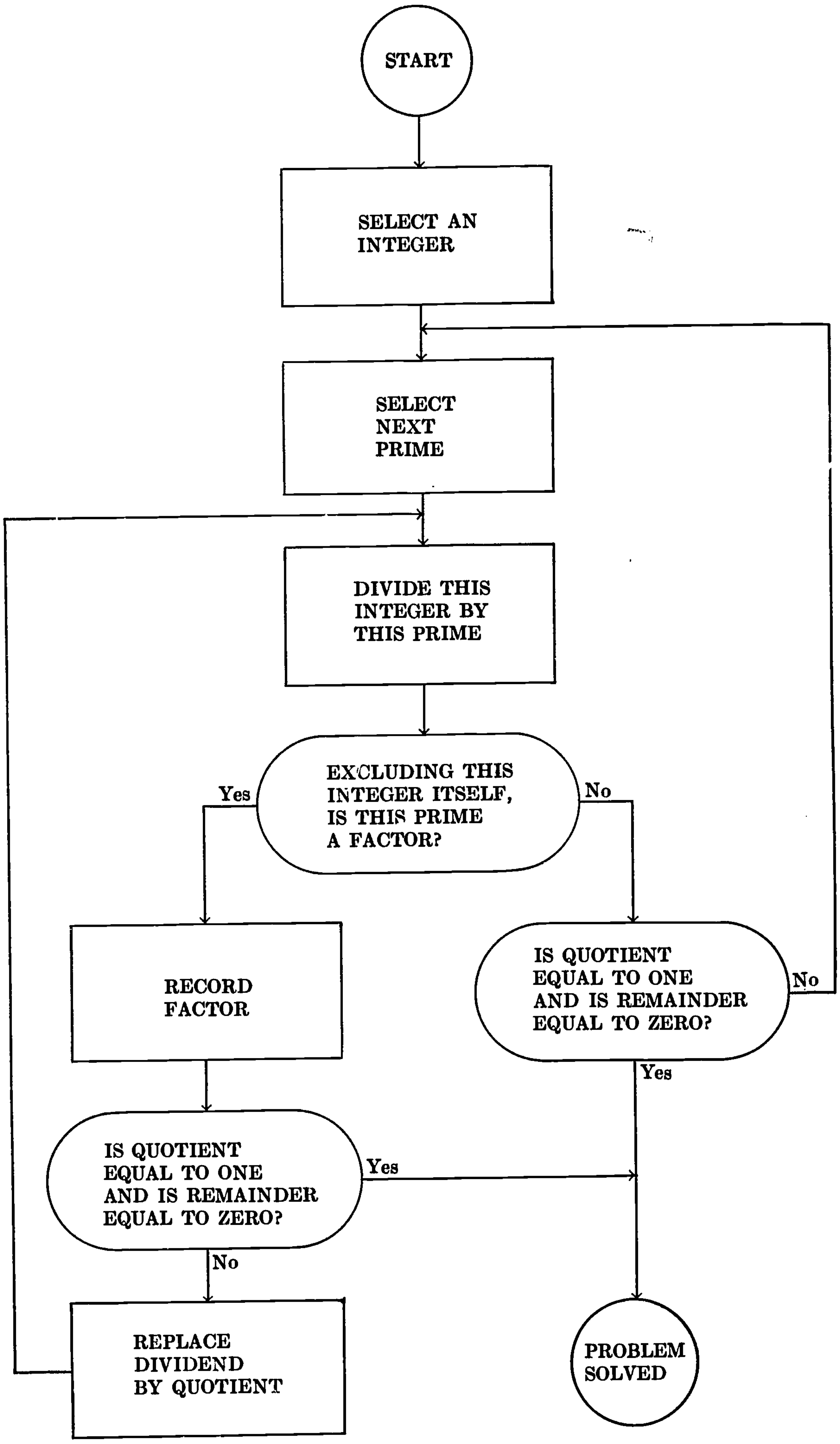


Fig. 3-4. Finding All the Prime Factors of an Integer

Euclid's algorithm is as follows:

1. Divide the larger integer, R_0, by the smaller integer, R_1, to obtain a remainder, R_2.
2. Divide R_1 by R_2 and obtain a remainder, R_3.
3. Divide R_2 by R_3 and obtain a remainder, R_4.
4. Continue until any remainder equals zero.

The last divisor just used is the greatest common divisor.

Suppose the greatest common divisor of 24 and 36 is wanted. Then, using Euclid's algorithm:

$$\begin{array}{rll} & 1 & \\ 24 & \overline{)36} & \\ & \underline{24} & 2 \\ & 12 & \overline{)24} \\ & & \underline{24} \\ & & 0 \end{array}$$

The greatest common divisor is 12.

Find the greatest common divisor of 17 and 2, and of 11 and 3:

$$\begin{array}{rll} & 8 & \\ 2 & \overline{)17} & \\ & \underline{16} & 2 \\ & 1 & \overline{)2} \\ & & \underline{2} \\ & & 0 \end{array} \qquad \begin{array}{rlll} & 3 & & \\ 3 & \overline{)11} & & \\ & \underline{9} & 1 & \\ & 2 & \overline{)3} & \\ & & \underline{2} & 2 \\ & & 1 & \overline{)2} \\ & & & \underline{2} \\ & & & 0 \end{array}$$

Here the greatest common divisor, in each case, is 1, as might be expected when two primes are used.

The algorithm for finding the greatest common divisor can be expressed as follows:

$$\begin{array}{rlllll} & Q_1 & & & & \\ R_1 & \overline{)R_0} & & & & \\ & \underline{x_1} & Q_2 & & & \\ & R_2 & \overline{)R_1} & & & \\ & & \underline{x_2} & Q_3 & & \\ & & R_3 & \overline{)R_2} & & \\ & & & \underline{x_3} & Q_4 & \\ & & & R_4 & \overline{)R_3} & \\ & & & & \underline{x_4} & \\ & & & & \text{etc.} & \end{array}$$

That is, each cycle can be represented by:

$$\begin{array}{rl} & Q_n \\ R_n & \overline{)R_{n-1}} \\ & \underline{x_n} \\ & R_{n+1} \end{array}$$

where n is related to the number of the cycle and, of course, increases as the process continues. When the remainder, R_{n+1}, becomes zero, the greatest common divisor will have been found and it is $R_{n+1-1} = R_n$. Specifically, on the fourth cycle, the above would be:

$$\begin{array}{r} Q_4 \\ R_4 \overline{)R_{4-1}} \\ \underline{x_4} \\ R_{4+1} \end{array} \quad \text{equals} \quad \begin{array}{r} Q_4 \\ R_4 \overline{)R_3} \\ \underline{x_4} \\ R_5 \end{array}$$

If R_{n+1}, or R_{4+1} for this cycle is zero, the greatest common divisor is $R_{n+1-1} = R_n$, or $R_{4+1-1} = R_4$. The flow diagram for this algorithm is shown in Fig. 3-5.

Example 6. Finding the Greatest Common Divisor of More Than Two Integers

Knowing Euclid's algorithm for finding the greatest common divisor of two integers, it is easy to apply the technique to more than two integers. Suppose the greatest common divisor of more than two integers is desired; for example, determine the greatest common divisor of 36; 114; 570; and 33. Choose any pair to begin with:

$$36 \text{ and } 114 \qquad \begin{array}{r} 3 \\ 36 \overline{)114} \\ \underline{108} \\ 6 \end{array} \quad \begin{array}{r} 6 \\ \overline{)36} \\ \underline{36} \\ 0 \end{array}$$

The greatest common divisor of this pair is 6. Use 6 and the next integer and continue as above:

$$6 \text{ and } 570 \qquad \begin{array}{r} 95 \\ 6 \overline{)570} \\ \underline{570} \\ 0 \end{array}$$

The greatest common divisor is still 6. Use 6 and the next integer and continue:

$$6 \text{ and } 33 \qquad \begin{array}{r} 5 \\ 6 \overline{)33} \\ \underline{30} \\ \text{GCD} \rightarrow 3 \end{array} \quad \begin{array}{r} 2 \\ \overline{)6} \\ \underline{6} \\ 0 \end{array}$$

Therefore, 3 is the greatest common divisor of 36; 114; 570; and 33.

References

Further information on the processes discussed in the foregoing text can be found in the following references, which are identified in the Bibliography, Appendix C: 16, 71, P6.

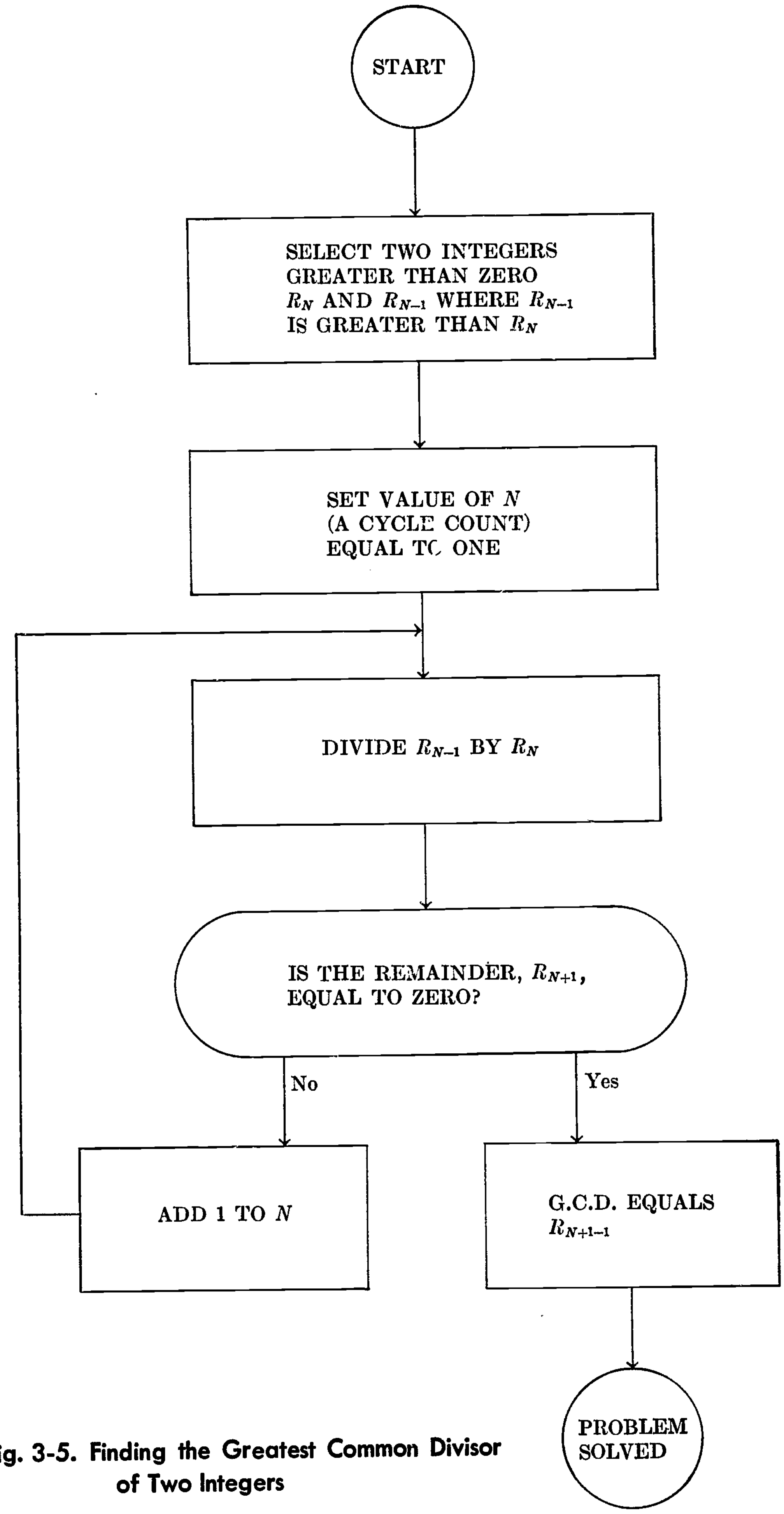


Fig. 3-5. Finding the Greatest Common Divisor of Two Integers

EXERCISES

Exercises Based on Examples 1, 2, and 3

1. Use the method shown in Example 2 to multiply (a) 16 × 42; (b) 123 × 123; (c) 421 × 342.
2. What is the minimum number of additions necessary to multiply 197 × 236 by the method of (a) Example 1? (b) Example 2?
3. In Example 2 three methods are shown for doing the same problem. What is the effect on the methods if the numbers to be multiplied contain zeros? For example, how does (a) 1768 × 2496 compare with multiplications such as (b) 1768 × 2006, or (c) 1708 × 2006?
4. Code Example 1 in machine language as shown in Chapter 2.
5. Consider the problem of coding Example 2 in machine language. What are some of the "tricky" sequences of instructions that would have to be written? Can you think of some additions to the operations that COMPIAC can perform that would make the operations more convenient to program?
6. Use the method shown in Example 3 for (a) 210 ÷ 30; (b) 152 ÷ 41.
7. Does the subtraction method of division work for 42 ÷ −5? Extend the flow diagram for Example 3 so that signed numbers may be used.
8. In Example 2 a "fast" method (Method III) of multiplication is illustrated. Draw a flow diagram for this method.

Exercises Based on Examples 4, 5, and 6

9. Is 1237 a prime? Draw a flow diagram showing how an organized iterative approach leads to a decision.
10. Find the prime factors of (a) 525; (b) 264; (c) 265; (d) 267.
11. In searching for the divisors of 1237, all successive primes less than 1237 could be used. This is a waste of time, however. Starting with 2, 3, 5, 7, etc., what is the criterion for terminating the search and being confident that the number is prime? Incorporate this feature into the flow diagram for Exercise 1.
12. Find the greatest common divisor of (a) 32 and 56; (b) 28 and 65; (c) 114 and 34; (d) 108, 118, and 30.
13. In Exercise 12 (d), does the selection of the first pair affect the complexity of the computation?

ANSWERS

2. (a) 197; (b) 11.
5. One of the "tricky" sequences would be a method of shifting the digits to the left which is the same as multiplying by 10, and doing this at the correct time. Some additional operation codes that would increase the capability of COMPIAC are *multiply, divide, shift,* etc. Computers can perform other basic operations than those discussed in Chapter 2.
7. No.
9. Yes.
10. (a) 3·5·5·7; (b) 2·2·2·3·11; (c) 5·53; (d) prime.

11. The search can be terminated when all the primes (p) not exceeding the square root of the number (n) have been used as trial divisors. If there has been no division without a remainder, n is prime.

12. (a) 8; (b) 1; (c) 2;(d) 2.

Part II. Approximations

Before discussing the use of repetitive processes in mathematical calculations, it is important to mention a few things about approximation. Life, in general, is a matter of making approximations. This is particularly true of making measurements. Measurements of physical objects depend on the comparison of the object with some sort of scale—a ruler or a micrometer, for example—either directly or indirectly. If an object is measured with a ruler divided into tenths of an inch, the result will be accurate only to the nearest tenth of an inch, although some good approximating can be done to less than one-tenth of an inch. For example, if measurement with such a ruler results in the value 3.4, the exact value is probably somewhere in the range between 3.35 and 3.45. The 3.4 is merely an approximation of the true value. In other words, it is a number which approximates another number. An important question is thus introduced: If in calculating, approximations of numbers are used instead of the exact numbers, how does this affect the result?

In using computers, it is very important to consider the errors that may occur in the result of a calculation when numbers which approximate other numbers are used. This is especially pertinent to the use of computers because problems solved by computers are usually very long and involve very large numbers of steps. Otherwise, they would not be solved by computers in the first place. Such problems, then, may involve thousands of arithmetic steps, and when numbers that are not exact are used, the total build-up of small errors may cause a serious error in the result.

Consider the following three numbers.

$$\pi \quad 3.1416 \quad 3.14$$

These three numbers are different, and either 3.14 or 3.1416 might be used to represent π in calculating, depending on practical considerations. In other words, 3.14 and 3.1416 are numbers which approximate π. Other examples are the replacement of $\sqrt{2}$, $\sqrt{3}$, $\frac{1}{3}$, $\frac{2}{3}$ with numbers such as 1.414, 1.732, 0.33333334, 0.667, respectively. The choice of a number to replace another number depends on how many digits are required in the calculation process to insure the required accuracy in the result.

A given digital computer can accept numbers up to a certain maximum size only. This maximum may be eight, ten, or some other number of digits, depending on the particular computer.

Accordingly, if digital methods are used for calculating and the arithmetic register of the computer can accommodate only ten digits, some fractions must be approximated. Using decimals, $\frac{1}{3}$ might be approximated either by

0.3333333333 or by 0.3333333334. Both figures are only slightly different from $\frac{1}{3}$, but the difference is important and may lead to difficulty.

To show what problems may develop when numbers that are slightly inaccurate are used, suppose an imaginary computer can handle only two decimal digits. In this case either 0.33 or 0.34 may be used as an approximation of $\frac{1}{3}$.

$$\text{Now } \frac{1}{3} = \frac{100}{300}, \text{ and } 0.33 = \frac{33}{100} = \frac{99}{300}.$$

Thus $\frac{1}{3}$ and 0.33 differ by $\frac{1}{300}$. However, using the same procedure, $\frac{1}{3}$ and 0.34 are seen to differ by $\frac{2}{300}$; so 0.33 is closer to $\frac{1}{3}$ than is 0.34.

Here is another example. One two-digit approximation of $\frac{2}{3}$ is 0.66; another is 0.67. Which is better?

$$\frac{2}{3} = \frac{200}{300}, \qquad 0.66 = \frac{198}{300}, \qquad 0.67 = \frac{201}{300}.$$

Thus 0.67 is a better approximation of $\frac{2}{3}$ than is 0.66.

To better understand some of the difficulties in using approximations, add $\frac{1}{3}$ to itself, and also add the "best" two-digit approximation for $\frac{1}{3}$, as above, to itself. Compare the result:

$$\tfrac{1}{3} + \tfrac{1}{3} = \tfrac{2}{3} \qquad \text{with} \qquad 0.33 + 0.33 = 0.66.$$

Now 0.33 differs from $\frac{1}{3}$ by $\frac{1}{300}$, but 0.66 differs from $\frac{2}{3}$ by $\frac{2}{300}$. Thus, the addition of the "best" approximations of $\frac{1}{3}$ to itself, does not produce the "best" approximation of $\frac{2}{3}$.

Notice the effect if the process is continued one more step:

$$\frac{2}{3} + \frac{1}{3} = \frac{3}{3} = 1.0 \qquad \text{but} \qquad 0.66 + 0.33 = 0.99.$$

Now, 0.99 is used to approximate 1.0, and the difference is $\frac{1}{100} = \frac{3}{300}$. This result is a poorer approximation than the original one, and the error is building up as the process continues.

Multiply $\frac{1}{3} \times \frac{1}{3} = \frac{1}{9}$. One approximation would be $0.33 \times 0.33 = 0.1089$. The result, 0.1089, is a four-digit number. Suppose it is possible to retain only two-digit numbers in the imaginary computer after the product is formed. There are two choices: (1) Discard the last two digits of the product 0.1089 and use 0.10 as the approximation of the product of $\frac{1}{3} \times \frac{1}{3} = \frac{1}{9}$. (Such a process is called *truncation*, in this case from four digits to two digits.) (2) Round off the number 0.1089 to the number 0.11. In both cases new approximations are being introduced.

If 0.10 is used as the approximate product, the difference is $\frac{1}{90}$ because

$$0.10 = \frac{1}{10} = \frac{9}{90}, \quad \text{and} \quad \frac{1}{9} = \frac{10}{90}.$$

However, if 0.11 is used as the approximate product, the difference is only $\frac{1}{90}$

because

$$0.11 = \frac{11}{100} = \frac{99}{900}, \quad \text{and} \quad \frac{1}{9} = \frac{100}{900}.$$

This is smaller than $\frac{1}{300}$, the difference between $\frac{1}{3}$ and its approximation, 0.33.

Thus, it appears that some repetitive calculating with approximations will increase the error and some will decrease the error. In some instances when the automatic digital computer is used in calculating with approximations, a detailed analysis of the kinds of errors that may possibly occur is necessary in preparing the problem for the machine. If it is known what kind of errors are likely to occur, it is usually possible to arrange the calculating procedure in such a way that the total error in the result remains smaller than some prescribed value. In other words, even though there may be approximation errors in the calculations, the results will still be useful, since the total error will not exceed some specified limit.

All this is designed to point out some of the difficulties that arise from the necessity of using approximations, particularly in repetitive computation. A full discussion of approximation, truncation, and round-off is beyond the scope of this publication, but it is an important aspect of numerical analysis.

The following paragraphs show how the effects of using approximations can be determined. Suppose the value 0.001 is recorded in an operation and it is known that the value is not exact—it may vary by 0.0005 in either direction, larger or smaller. This may be expressed as 0.001 ± 0.0005 or as 0.001(5). Either expression defines the range within which the true value lies. In the above case the upper limit of the range is 0.0015, and the lower limit is 0.0005.

For convenience in manipulation, the range may be expressed as follows.

$$\begin{bmatrix} 0.0015 \\ 0.0005 \end{bmatrix}$$

Consider the problem of adding 0.001(5) to 0.035(2). This, also, can be expressed as follows.

$$\begin{bmatrix} 0.0015 \\ 0.0005 \end{bmatrix} + \begin{bmatrix} 0.0352 \\ 0.0348 \end{bmatrix} = \begin{bmatrix} 0.0367 \\ 0.0353 \end{bmatrix}$$

We see by this example that if one adds the number 0.001 (which may be in error by as much as 5 in the fourth place to the right of the decimal) to the number 0.035 (which may be in error by as much as 2 in the fourth place to the right of the decimal), the result may be any where in the range of 0.0353 to 0.0367. Consequently, in this kind of example, one should attach only limited significance to the result beyond the second decimal place because the answer is "accurate" to two places only.

The error is even more pronounced in multiplication, as shown by the following example in which the number 0.001(5) is multiplied by the number 0.035(2).

$$\begin{bmatrix} 0.0015 \\ 0.0005 \end{bmatrix} \times \begin{bmatrix} 0.0352 \\ 0.0348 \end{bmatrix} = \begin{bmatrix} 0.00005280 \\ 0.00001740 \end{bmatrix}$$

In other words, the true value of the product is somewhere between 0.00001740 and 0.00005280.

If one had assumed in a given problem that the original values (0.001 and 0.035, for example) were exact, when actually they were subject to variations in the fourth place to the right of the decimal point by 5 and by 2 respectively, the resulting error might become very important, since the "exact" result equals 0.000035.

The paragraphs above have suggested certain difficulties which may arise as a consequence of using certain numbers as approximations for other numbers. On the other hand, approximations are required. As stated before, the development and use of such numbers as are used to represent other numbers is a very important part of numerical analysis, because essentially "the answer is no better than the numbers which were used to produce it."

References

For more information on the subject of approximations, consult the following references, which are identified in the Bibliography, Appendix C: 21, 32, 34, 47, 52, 56, 60.

EXERCISES

1. Find the measures of the area of a circle with a radius of 21 units by using: (a) $3\frac{1}{7}$, and (b) 3.14 as approximations of π. (c) What is the difference between the two measures of area?
2. One approximation for π is 3.14159265. What is the difference between 3.14159265 and $3\frac{1}{7}$ approximated to the same number of decimal places?
3. As explained in Part II of Chapter 3, what is the upper and lower limit of the range indicated by the following expression: 0.004(5)?
4. Find the upper and lower limits of the following addition and subtraction problems:
   (a) $0.003(5) + 0.024(3) =$
   (b) $0.537(2) + 0.423(4) =$
   (c) $1.000(5) - 0.987(2) =$
   HINT: Are the upper and lower limits found in the same way as in addition?
   (d) $2.856(1) - 1.742(3) =$
5. Find the upper and lower limits of the following products:
   (a) $0.002(5) \times 0.045(2) =$
   (b) $0.34(3) \times 0.21(5) =$
   (c) $1.2(5) \times 1.2(5) =$

ANSWERS

1. (a) 1386 sq. units, (b) 1384.74 sq. units. (c) 1.26 sq. units.
2. 0.00126449.
3. Upper limit, 0.0045; lower limit, 0.0035.

4. (a) $\begin{bmatrix} 0.0278 \\ 0.0262 \end{bmatrix}$ (b) $\begin{bmatrix} 0.9606 \\ 0.9594 \end{bmatrix}$ (c) $\begin{bmatrix} .0137 \\ .0123 \end{bmatrix}$ (d) $\begin{bmatrix} 1.1144 \\ 1.1136 \end{bmatrix}$

5. (a) $\begin{bmatrix} .00011300 \\ .00006720 \end{bmatrix}$ (b) $\begin{bmatrix} .073745 \\ .069085 \end{bmatrix}$ (c) $\begin{bmatrix} 1.5625 \\ 1.3225 \end{bmatrix}$

Part III. Further Examples of Numerical Calculation

The following examples involve procedures that are not only repetitive but may also introduce errors because approximate numbers are used. Additional illustrative examples are given in Appendix B.

Example 1. Evaluating an Algebraic Function

It is frequently necessary to evaluate an expression, such as the one below, for several values of x.

$$y = 6x^4 + 4x^3 - 5x^2 + 6x + 4$$

It is possible, of course, to start with a given value of x, develop the powers of x which are required, perform the necessary multiplications by the coefficients, and finally produce the sum. The following steps might be called for by this process:

1. Select x
2. Multiply x by x and store x^2
3. Multiply x^2 by x and store x^3
4. Multiply x^3 by x and store x^4
5. Multiply x by 6 and store $6x$
6. Multiply stored x^2 by 5 and store $5x^2$
7. Multiply stored x^3 by 4 and store $4x^3$
8. Multiply stored x^4 by 6 and store $6x^4$
9. Add $6x^4$
10. Add $4x^3$
11. Subtract $5x^2$
12. Add $6x$
13. Add 4.

Another possible approach to the problem is to rewrite the expression

$$y = 6x^4 + 4x^3 - 5x^2 + 6x + 4$$

as

$$y = x(6x^3 + 4x^2 - 5x + 6) + 4.$$

The expression inside the parentheses can be rewritten as:

$$x[6x^2 + 4x - 5] + 6.$$

Again, the expression inside the brackets can be rewritten as:

$$x\{6x + 4\} - 5.$$

Once more, the expression inside the braces can be rewritten as:

$$x\langle 6 \rangle + 4.$$

By reassembling the above expressions as follows:

$$y = x(x\,[x\{x\langle 6 \rangle + 4\} - 5\,] + 6) + 4$$

the order of operations would be:

1. Select x
2. Multiply by 6
3. Add 4
4. Multiply by x
5. Subtract 5
6. Multiply by x
7. Add 6
8. Multiply by x
9. Add 4.

In this way the process is completed in fewer steps and can be reduced to an iterative procedure, which is adaptable either to a digital computer or to hand calculations.

Tables 3-1 and 3-2 show the effect of using the two methods described above as far as the introduction of error is concerned. This is done by using three values of x—2, 2.05, and 2.01—for each method to allow comparison in the result. The error is introduced because intermediate results may be rounded off. For example, to obtain the results in the columns headed "Round-off," the digit in the second decimal place in calculating was increased by 1 when the digit in the third decimal place was 5 or greater; that is, 0.267 would be rounded to 0.27, whereas 0.264 would be left as 0.26.

Where the round-off error is not involved, as in the first columns of both

TABLE 3-1
FIRST METHOD

Operation	For $x = 2$	For $x = 2.05$		For $x = 2.01$	
	No round-off	Round-off	No round-off	Round-off	No round-off
1. Select x	2	2.05	2.05	2.01	2.01
2. $x \cdot x = x^2$	4	4.20	4.2025	4.04	4.0401
3. $x^2 \cdot x = x^3$	8	8.61	8.615125	8.12	8.120601
4. $x^3 \cdot x = x^4$	16	17.65	17.66100625	16.32	16.32240801
5. $x \cdot 6 = 6x$	12	12.30	12.30	12.06	12.06
6. $x^2 \cdot 5 = 5x^2$	20	21.00	21.0125	20.20	20.2005
7. $x^3 \cdot 4 = 4x^3$	32	34.44	34.460500	32.48	32.482404
8. $x^4 \cdot 6 = 6x^4$	96	105.90	105.96603750	97.92	97.93444806
9. $6x^4$	96	105.90	105.96603750	97.92	97.93444806
10. $+4x^3$	128	140.34	140.42653750	130.40	130.41685206
11. $-5x^2$	108	119.34	119.41403750	110.20	110.21635206
12. $+6x$	120	131.64	131.71403750	122.26	122.27635206
13. $+4$	124	135.64	135.71403750	126.26	126.27635206

TABLE 3-2
SECOND METHOD

Operation	For $x = 2$	For $x = 2.05$		For $x = 2.01$	
	No round-off	Round-off	No round-off	Round-off	No round-off
1. Select x	2	2.05	2.05	2.01	2.01
2. $\cdot 6$	12	12.30	12.30	12.06	12.06
3. $+4$	16	16.30	16.30	16.06	16.06
4. $\cdot x$	32	33.42	33.4150	32.28	32.2806
5. -5	27	28.42	28.4150	27.28	27.2806
6. $\cdot x$	54	58.26	58.250750	54.83	54.834006
7. $+6$	60	64.26	64.250750	60.83	60.834006
8. $\cdot x$	120	131.73	131.71403750	122.27	122.27635206
9. $+4$	124	135.73	135.71403750	126.27	126.27635206

tables, the results are, of course, the same for both methods. Likewise, the results are the same in both tables for $x = 2.05$ and $x = 2.01$ in the columns headed "No round-off."

Notice, however, the difference in the results for a given column depending on whether or not round-off is used. For the first method, when $x = 2.05$, the difference to two decimal places is as given below.

$$\begin{array}{r} 135.71 \\ -\underline{135.64} \\ .07 \end{array}$$

For the second method, the difference to two decimal places is as given below.

$$\begin{array}{r} 135.71 \\ -\underline{135.73} \\ -\quad .02 \end{array}$$

Similarly, when $x = 2.01$, the first and second methods show different values of round-off error.

First method	Second method
126.28	126.28
$-$126.26	$-$126.27
.02	.01

It can be seen that simple expressions are desirable not only because they are more easy to use. Because they require fewer operations, the errors that are introduced by round-off are not operated on so many times. As a result, when a simple process is used, the final answer is less inaccurate than the answer obtained by an extended process.

The above example showed errors that were comparatively small. However, in complex numerical calculations, which require many thousands or even millions of operations, the error introduced, due to round-off, may be of very great importance.

A particularly sensitive situation is such as the following:

$$Y = \frac{L}{(M - N)}.$$

If M and N each are the results of many previous calculations, the Y might have a different algebraic sign than it should have if M and N were very nearly the same size and the errors had been "just right." Also, such errors might produce the situation where an attempt would be made to divide by zero. For example, suppose the "true" value of M is 500 and the "true" value of N is 499. That is, assume that there were no round-off errors in the calculations which produced M and N.

However if, because of error introduced by round-off, M were 499.57, that is, "too small" by 0.43; and N were 499.57, that is, "too large" by 0.57, the result would be

$$(M - N) = 0$$

a condition which, in the subsequent computation of Y, could not be taken care of in the usual manner.

NOTE. Computers can be programmed to check for the zero condition, for example, before any division step where it might possibly occur. In the COMPIAC compiler, such a statement might be

$$(M - N) = 0\text{: STOP. CONTINUE.}$$

Example 2. A Method for Finding Square Root

If A is an integer, and $x_0 \neq \sqrt{A}$ is an approximation for the square root of A, then A/x_0 will be either larger or smaller than the square root of A, depending on whether x_0 is smaller or larger than the desired root. If x_0 is smaller than the root, for example, then the quotient A/x_0 will be larger than the root. The average of x_0 and A/x_0 will be a better approximation of the square root of A than either x_0 or A/x_0. Conversely, if x_0 is larger than the true root, a similar relation holds. Thus we may improve an initial estimate of the square root of a number, A, by successively dividing it by an estimate of the root and averaging this quotient with the estimate to get a new estimate. The algorithm is as follows:

1. Make an estimate of the square root of the number.
2. Divide the number by the estimate.
3. Obtain the average of the divisor just used and the quotient just obtained.
4. Divide the number by this average.
5. Repeat steps 3 and 4 until the desired accuracy is obtained.

As an example, use this method to find the square root of 25.

1. Let 2 be the first estimate.

2. Divide: $2\overline{)25}$ = 12.5

3. Average: $\begin{array}{r} 12.5 \\ +\ 2.0 \\ \hline 14.5 \end{array} \quad \begin{array}{r} 7.25 \\ 2\overline{)14.50} \end{array}$

4. Divide: $\begin{array}{r} 3.45 \\ 7.25\overline{)25.00} \end{array}$

5. Average: $\begin{array}{r} 3.45 \\ +7.25 \\ \hline 10.70 \end{array} \quad \begin{array}{r} 5.35 \\ 2\overline{)10.70} \end{array}$

6. Divide: $\begin{array}{r} 4.67 \\ 5.35\overline{)25.00} \end{array}$

7. Average: $\begin{array}{r} 5.35 \\ +4.67 \\ \hline 10.02 \end{array} \quad \begin{array}{r} 5.01 \\ 2\overline{)10.02} \end{array}$

8. Divide: $\begin{array}{r} 4.99 \\ 5.01\overline{)25.00} \end{array}$

Notice that the process is continuous and has no stopping point, and that there is no indicated way of deciding how accurate the estimate has become after n steps. One way to solve both problems is to include certain steps after each division in which the new quotient is squared. The squared value can then be compared with A, and if the difference between A and the quotient squared is less than a specified number, the process is stopped.

A question may occur on how to start the process: How is the first estimate obtained? This may be accomplished by various methods, one of which is to take one-half of A as an initial estimate. The diagram in Fig. 3-6 indicates the procedure for calculating the square root of a number in such a way that the difference between the original number and the square of some later estimate is less than 0.0001. The steps are as follows:

1. Take one-half of the number as the first estimate of the root.
2. Divide the number by this estimate.
3. Average the divisor just used and the quotient just obtained.
4. Multiply the average obtained by itself (square the average).
5. Subtract the product thus obtained from the number. If the absolute value of the difference is less than 0.0001, go to Step 8; otherwise, go to Step 6.
6. Use the average value just obtained as the next estimate.
7. Repeat Steps 2, 3, 4, and 5.
8. Stop. Last average is desired root.

It may be noticed that Steps 1 and 2 of this procedure are redundant, since on the first cycle Step 2 will always give a result of 2. These two steps, however, serve to get the procedure started, and the rest of the steps produce the approximate square root.

NOTE. The above procedure can be expressed in another way, as follows: If A is the number the square root of which is desired and x_0 is an approximation of the square root, x_1, an improved approximation can be determined by

$$x_1 = (x_0/2) + (A/2x_0).$$

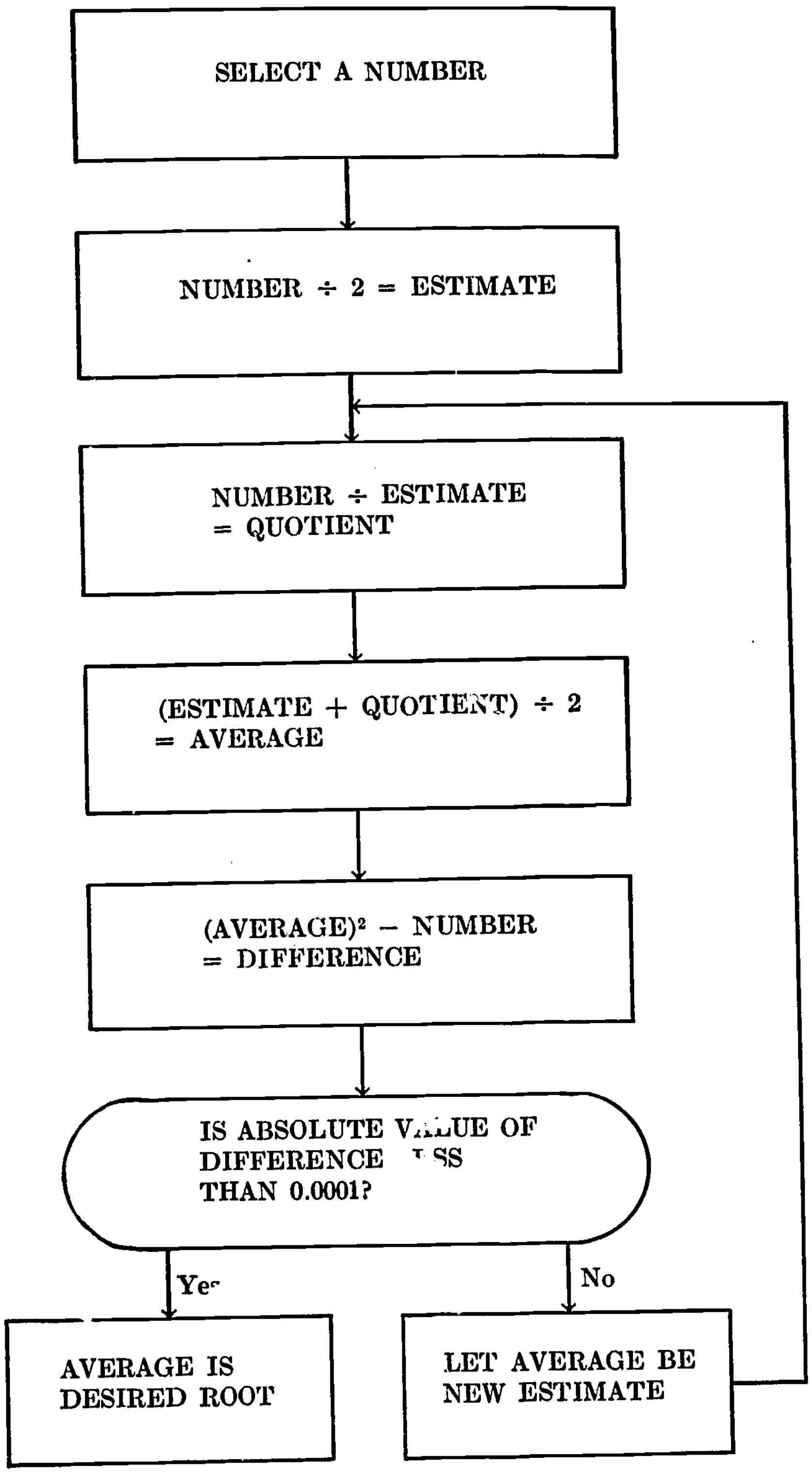


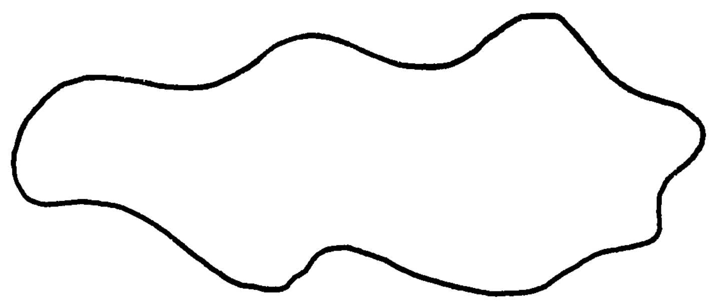
Fig. 3-6. Method of Approximating the Square Root of a Number

Example 3. Measuring the Area of An Irregular Figure

For solving certain problems there is an iterative method which usually depends on the availability of a computing machine, since very many iterations are often required. Thus, this method may not be practical for calculating by hand.

In this method the principle of operation is quite unusual and is based on the construction of a mathematical model of the problem to "let nature take its course."

For example, suppose it is desired to determine the area of an irregularly shaped figure, such as the one below:



Given enough time, the following method would provide an answer.

1. Cut out a piece of metal of known size, say 2 by 2 inches. Call this piece of metal A.
2. Cut out a piece of metal shaped like the figure the area of which is to be determined. Call this piece B.
3. To each piece of metal attach a counter which advances by 1 each time a raindrop strikes the metal piece.
4. Suspend the two pieces of metal in such positions that raindrops are as likely to strike one piece as the other.
5. Let nature take its course for a while. This assumes that raindrops fall in a random manner; that is, knowing where all previous raindrops have fallen, it is not possible to predict where the next one will fall.
6. Read the counters and substitute the counts in the expression:

$$\frac{\text{4 square inches}}{\text{Counts on } A} = \frac{\text{Area of unknown figure}}{\text{Counts on } B}$$

If raindrops fall at random over a space containing two pieces of metal, the probability that a raindrop will hit one part of the space is the same as the probability that a raindrop will hit any other part of the space. In other words, the probability that each part of the space will be hit by a raindrop gets closer and closer to certainty the longer the rain continues to fall on the space.

The number of raindrops hitting a given part of the space (one of the metal pieces) is proportional to the area of the given part.

If the process does not continue for a sufficiently long time before the counts are taken and substituted in the above expression, the answer may not be very accurate. On the other hand, after a certain period of time the answer may not improve very much either, since all of the area will have been affected.

This kind of process can be performed with a digital computer, using as a substitute for the raindrops a list of numbers called *random numbers*. An important property of random numbers is that, given one or more of them, it is not possible to predict what the next one will be. In other words, they are dependent on chance. A computer can generate numbers which are very much like random numbers, but since they are computed by a program, they are predictable and not really dependent on chance. They are called *pseudo-random numbers* and are used just as if they were really random numbers.

Methods based on randomness are called *Monte Carlo methods*.

Example 4. Finding the Ratio of Two Areas

The following example is similar to the one above. Take an area nine units wide on each side and divide it as shown by the heavy line in Fig. 3-7. Call the part above the dark line A and the part below the dark line B. The problem is as follows: What fraction of the total area is Part A? The diagram in Fig. 3-8 shows one possible plan for solving the problem.

An interesting alternate method for solving this problem involves the use of a pair of nine-sided dice. Let one die represent the Y axis and the other die represent the X axis. Assume that for each throw of the dice any number on each die is just as likely to come up as any other number; in other words, the numbers will come up in random fashion. The problem-solving procedure to follow is to roll the dice and put marks in the appropriate squares according to the values that come up. Continue to roll the dice and mark squares while keeping count of the total number of throws. After a number of throws, determine the ratio of the number of marks in Part A to the total number of throws. This will be an approximation of the answer.

An important part of such a process is knowing when to stop. One way of determining this is to calculate the ratio after a number of throws, then continue to throw, and finally calculate again. If the two values of the ratio are substantially different, try it for a while longer and calculate again. At the time when two successive values are sufficiently alike so that their difference is not

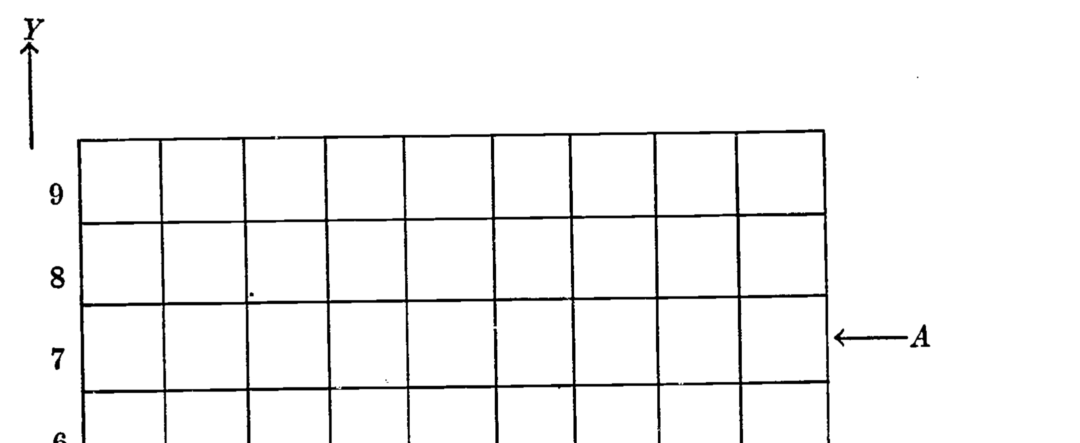


Fig. 3-7. What Fraction of Total Area Is the Area of A?

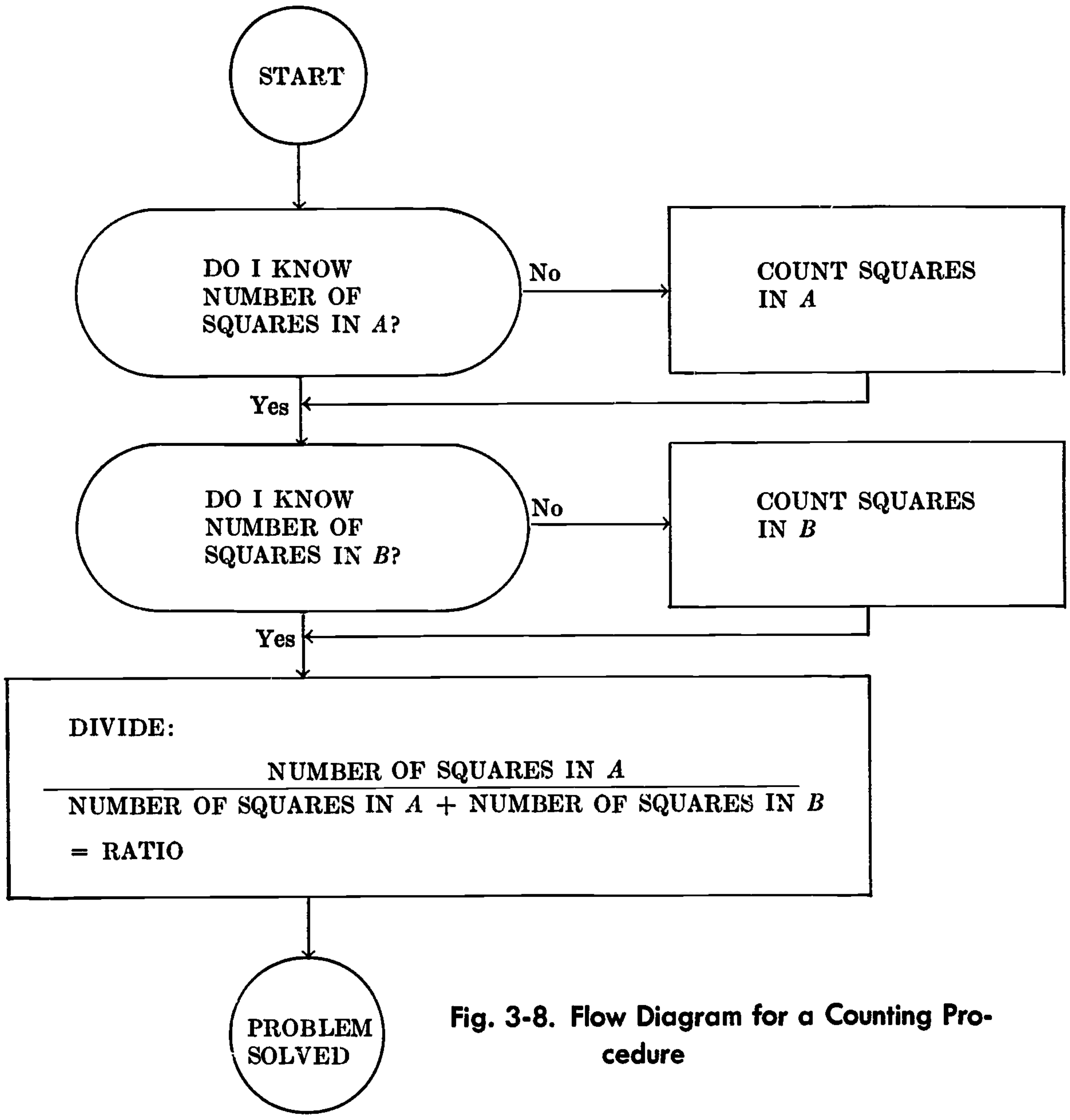


Fig. 3-8. Flow Diagram for a Counting Procedure

significant in view of the particular application, stop the process. This procedure can also be used with regular dice and a square six-units wide on each side. This method may give an answer that is not as accurate as the square-counting method, although for a given application it can be accurate enough. It will also require a great deal of dice-throwing. However, as stated above, computers can be made to do the equivalent of dice-throwing and to do it very rapidly. Of course, computers can count squares rapidly too. The use of random-number methods is usually justified only when no better direct methods exist. The above examples are merely intended to suggest a problem-solving technique.

The random-number method can be extended to the level where the model is not a piece of metal nor even a graph but a mathematical expression. For example, one can imagine that a circle is drawn inside the above 9 by 9 square and the dice-throwing procedure followed. The procedure would work to some extent, but in the squares through which the circle passed it would not be clear whether the mark should be counted as being inside or outside the circle. Therefore, the count would be rough—or in error. One way of overcoming this kind of difficulty would be to divide the square into smaller divisions so that the inside-or-outside question would pertain to a smaller fraction of the total area. It can be imagined that this fineness could be extended to a square with 100; 1,000; 10,000; or more subdivisions. This, of course, complicates the dice-throwing and record-keeping requirements.

Another approach is to select an X and a Y and determine, not graphically but mathematically, whether the point selected is inside or outside the circle. To visualize this approach imagine a square with a circle inside, such as shown in Fig. 3-9.

For this particular situation the equation of the circle, $X^2 + Y^2 = R^2$, translated to the interior of the square so that its center is at $X = 6$, $Y = 5$, is

$$(X - 6)^2 + (Y - 5)^2 = 3^2.$$

Now if the computer does the equivalent of rolling dice to locate X and Y, each between 0 and 9, both inclusive, of course the point can be considered to be inside or outside the circle. But instead of plotting the points, the pair of numbers can be substituted for the variables X and Y in the equation. If the result of the substitution is equal to or less than 9, the point is inside the circle; otherwise, it is outside.

The procedure, then, is as follows:

1. Select an X and a Y.
2. Try the selected X and Y in the equation.
3. Keep count of whether or not the point is inside or outside the circle.
4. Eventually compare the count with the total number of trials.

This process can be refined by using a mathematical square with 1,000 units per side, or with 10,000 units per side. The only difference in the requirements is in the size of the sets of random numbers which the computer generates.

In summary, then, this example shows how areas bounded by curves may be determined by an iterative process applied to mathematical models. Since the determination of areas bounded by curves or straight lines is closely related to

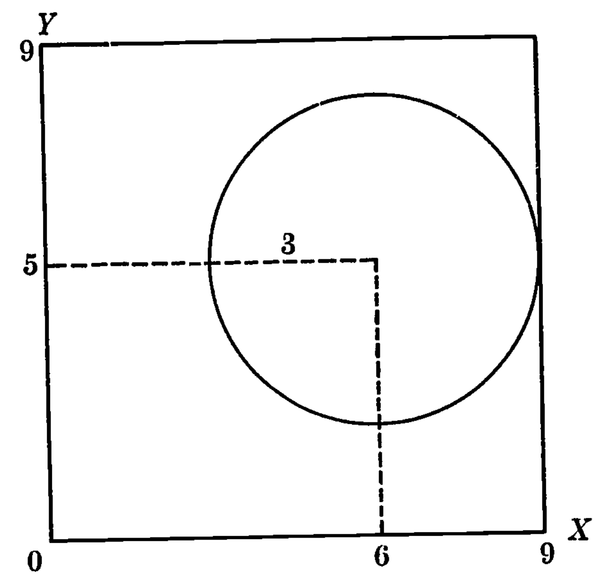


Fig. 3-9. Ratio of Circle Area to Square Area

the process of integration, it is not difficult to imagine extensions of Monte Carlo methods to such problems.

An interesting application of Monte Carlo methods is in the study of shields offering protection to people working near sources of atomic radiation, the question being: Will the radiation get through the shield? For a given sample shield, the distribution of its atomic and molecular components—that is, the number of each of the several types of atoms that make up the metal—would be known. For a given source of atomic radiation there would also be known characteristics. Also, the effect of one particle hitting another can be calculated, using relationships known to physicists. For example, if a particle from the atomic source (with a given mass, velocity, and direction of motion) strikes an atom in the shield (which also has a particular mass, velocity, and direction of motion), the resultant velocities and directions of motion can be calculated.

It is possible to simulate the path of a particle as it goes from collision to collision and see whether the particle will penetrate the shield or will be stopped. Monte Carlo methods can be used, for example, to select (by letting nature take its course): (1) the *initial direction* of a radiated particle as it first strikes the edge of the shield and (2) the *types of particles* in the shield—and their *velocities* and *directions of motion*—which the radiated particle will hit in successive collisions.

The procedure, then, is to select at random the direction of striking the shield and start the particle—certain characteristics of which are also determined at random—on its way, collision by collision. The result will be one of the following three possibilities: (1) The particle is stopped. (2) The particle penetrates the shield. (3) The particle bounces back toward its source. The process is then repeated, with another particle following another path, and so on.

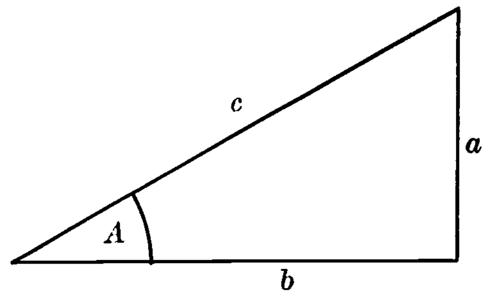
The ratio of the number of particles that penetrate to the number of particles that were started is a measure of the shield's effectiveness. Thousands of trials may be required before the ratio settles down. If too many particles get through in the simulation, the mathematical model of the shield may be changed either by giving it a new thickness or by giving it new chemical characteristics with a correspondingly different distribution of components. The simulation is now tried again for the necessary number of trials to obtain a new ratio, and the shields are compared.

It is not difficult to imagine other uses of such methods for problems where it is not possible to determine directly the effect of "nature" on the solution. Of course, a digital computer is quite important in this type of problem solving which involves such a great amount of arithmetic.

Example 5. Evaluating Trigonometric Relations

For many problems in mathematics and in engineering, the relationships between the sides of a right triangle are important, and this, of course, may suggest a general definition of trigonometry. It is interesting to consider some of the aspects of this field from a computer-oriented point of view. That is to say, if a computer is available, how can certain trigonometric problems be approached differently than if they were done by hand.

In a triangle there are the following six relationships:



$$\frac{a}{c} = \sin A; \quad \frac{b}{c} = \cos A; \quad \frac{a}{b} = \tan A; \quad \frac{c}{a} = \csc A; \quad \frac{c}{b} = \sec A; \quad \frac{b}{a} = \cot A.$$

It is not necessary to consider the last three functions in the same sense as the first three because, if any one of the first three can be obtained, the corresponding function of the last three can be determined simply by taking the reciprocal.

The advantage of this capability is that in using a digital computer it is desirable to avoid as much programming as possible, since not only does the initial programming take time but storage capacity is required to contain the program. Therefore, instead of preparing a computer program both for the sine and cosecant of an angle—and thereby using storage space for two programs—it may be advantageous to have but one program, say the sine, and to produce the cosecant from it.

The paragraphs below describe two methods of obtaining the trigonometric functions. One method calls for the expansion of a particular series; the other calls for the evaluation of a polynomial. Before discussing these methods, however, the idea of *reducing the number of necessary functions*, as discussed above, will be extended.

Reference to the triangle above shows that

$$\tan A = \frac{a}{b}$$

and that tan A is related to sin A and cos A by the following:

$$\frac{\sin A}{\cos A} = \frac{a/c}{b/c} = \frac{a}{b} = \tan A.$$

A similar manipulation is shown below.

Following the Pythogorean theorem,

$$a^2 + b^2 = c^2$$

and dividing by c^2

$$\frac{a^2}{c^2} + \frac{b^2}{c^2} = \frac{c^2}{c^2}$$

or

$$\left(\frac{a}{c}\right)^2 + \left(\frac{b}{c}\right)^2 = 1$$

or

$$\sin^2 A + \cos^2 A = 1.$$

Apparently, then, since

$$\cos A = \sqrt{1 - \sin^2 A},$$

it is necessary only to be able to obtain sine A because with this—and the ability to add, subtract, multiply, divide, and extract square root—the other five functions may be obtained without needing to use any more storage capacity than is required to obtain the sine of an angle and to obtain the square root of a number. And, since the square root program is usually included anyway for other calculations, the six trigonometric functions can be made available by merely writing a program for one of them.

However, in some instances it is necessary to have more than one function or to be extremely careful in using the program. Consider:

$$\sin^2 A + \cos^2 A = 1$$

then

$$\sin A = \sqrt{1 - \cos^2 A}.$$

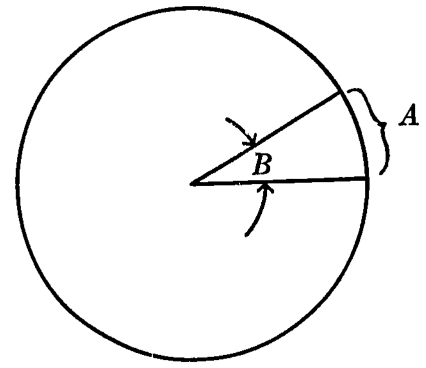
In attempting to compute sine A from cosine A by using the above function for small values of angle A, cosine A differs so slightly from 1.0 that for all practical purposes, considering a given number of significant figures, it will be the same as 1.0. For example, consider a table of natural trigonometric functions correct to four decimal places as given below.

Angle A	Cosine A	Sine A
$0°0'$	1.0000	0.0000
$0°1'$	1.0000	0.0003
$0°2'$	1.0000	0.0006
⋮	⋮	⋮
$0°9'$	1.0000	0.0026
$0°10'$	1.0000	0.0029

Obviously, using the above values for cosine A and computing sine A by the procedure suggested, the value 0.0000 would be obtained for sine A from $0°\,0'$ through $0°\,10'$, whereas the value for sine A actually ranges from 0.0000 through 0.0029 for small positive angles from $0°\,0'$ through $0°\,10'$.

Before discussing the methods of computing the sine, it should be pointed out that in many practical applications of the trigonometric functions it is advantageous to use the radian rather than the degree as the unit of angle measurement. The advantage is noted particularly when the applications are related to devices that may have circular motion (such as rotating shafts and generators) and oscillatory motion based on rotation (such as pistons, sewing machines, and lift pumps).

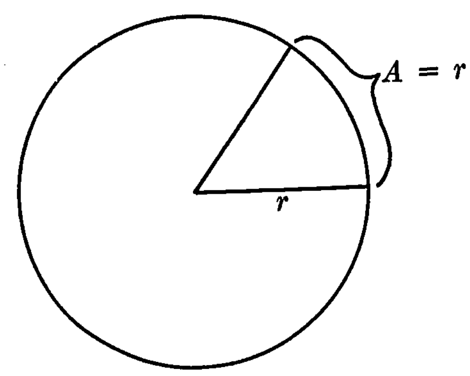
The radian, like the degree, is based on the division of a circle into a certain number of parts.



The arc, A, of the circle determines the size of the angle, B, and the degree system of angle measurement is based on the division of the circle's circumference into 360 equal parts. The degree, than, is defined as the angle determined, as above, by an arc, such as A, which is 1/360 of the circumference.

Of course, any other division of the circle would serve the same purpose, and the trigonometric functions would be unchanged. For example, one other system divides the circle into 6,400 parts, and the angle defined by the 1/6,400 part of a circle is called a mil. This method of division is widely used in military gunnery because it relates angular deflection to range.

The radian is based on still another system in which the arc of the circle, A, which defines the angle is equal to the radius, R, of the circle itself.



This means, then, that since the circumference of a circle is equal to $2\pi R$ and the radian system divides the circle into R parts, a radian is $1/R$th of a circle. Therefore, there are

$$\frac{2\pi R}{R} = 2\pi$$

radians in a circle, or

$$2\pi = 2 \times 3.1416 = 6.2832 \ldots \text{ radians.}$$

A radian, therefore, is equal to $360/6.2832 \ldots = 57° \, 17' \, 44.8''$ or $57.29577°$.

One way of calculating the sine of an angle, measured in radians, and therefore all the other functions as suggested above, is to substitute in the following series:

$$\sin x = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \cdots$$

As can be seen, this method lends itself to repetitive calculations because successive numerators can be developed merely by multiplying the last one by x^2, and the successive denominators can be developed by increasing the value of the factorial by the next two additional multiplications. For example, given x, the following steps would produce the numerators:

Step 1. Obtain x^2
Step 2. Multiply x by $x^2 = x^3$
Step 3. Multiply x^3 by $x^2 = x^5$
Step 4. Multiply x^5 by $x^2 = x^7$
etc.

In a similar fashion the denominators can be obtained as follows:

Step 1. Start with digit 1
Step 2. Add 1 to last digit $(1 + 1 = 2)$
Step 3. Add 1 to this last digit $(1 + 2 = 3)$
Step 4. Add 1 to this last digit $(1 + 3 = 4)$
etc., to produce the following series:
1, 2, 3, 4, 5, 6, 7, 8, 9, ...

With this series, consider the numbers to be arranged as follows:

$$1(2\cdot 3)(4\cdot 5)(6\cdot 7)(8\cdot 9)\ \ldots$$

Now follow this series of steps:

Step 1. Use the first number (1) as the first denominator
Step 2. Multiply the next two numbers $(2\cdot 3)$
Step 3. Multiply the result of Step 2 by the result of Step 1: $1(2\cdot 3)$
Step 4. Multiply the next two numbers $(4\cdot 5)$
Step 5. Multiply the result of Step 4 by the result of Step 3
etc.

Notice that each additional denominator is obtained by multiplying the next two numbers together and multiplying this result by the result obtained up to that time. In order to stop the process, it is possible to examine the contribution of a given term, say the eleventh, by subtracting the accumulated value obtained before the eleventh term was included from the value obtained after the value was included. For example:

$$\begin{array}{rl} 0.6283674 & \text{(total not including the eleventh term)} \\ \underline{-0.6283680} & \text{(total including the eleventh term)} \\ -0.0000006 & \text{(difference)} \end{array}$$

If the difference is small enough to be unimportant for the particular application, the process can be stopped.

The sine of an angle can be computed in other ways, based on the evaluation of a polynomial. This system depends on determining a polynomial with characteristics that, within certain limits, can be substituted for the sine function above. The following polynomial is taken from *Approximations for Digital Computers*, by Hastings (reference 32).

$$\sin\left(\frac{\pi}{2}\cdot x\right) = 1.570794852x - 0.645920978x^3 + 0.079487663x^5 - 0.004362476x^7$$

This particular polynomial may introduce errors into the sixth decimal place. Other polynomials, also listed in the same book, are available if greater accuracy is desired. The value of x is limited to the range of -1 to $+1$. This means, essentially, that the expression can be used for any value from 0 to 90 degrees, inclusive, since

$$\text{for } x = 1, \quad \frac{\pi}{2} \cdot x = \frac{\pi}{2} \cdot 1 = \frac{\pi}{2} \text{ or } 90^\circ$$

and

$$\text{for } x = 0, \quad \frac{\pi}{2} \cdot x = \frac{\pi}{2} \cdot 0 = 0^\circ.$$

For example, using $x = 1$, the above expression would have the following value:

$$\begin{array}{r} 1.570794852 \\ -0.645920978 \\ +0.079487663 \\ \underline{-0.004362476} \\ 0.999999061 \end{array}$$

which is a good approximation for the sine of 90 degrees.

The original expression is of the following form:

$$\sin\left(\frac{\pi}{2} \cdot x\right) = C_1x + C_3x^3 + C_5x^5 + C_7x^7$$

where the C's are the constants given above; x can be obtained for any number of degrees by first converting to the equivalent value of 90 degrees or less, and then expressing this value as a fractional part of 90 degrees. For example:

$$\text{for } 90^\circ, x = \frac{90}{90} = 1; \quad \text{for } 45^\circ, x = \frac{45}{90} = 0.5; \quad \text{for } 22.5^\circ, x = \frac{22.5}{90} = 0.25.$$

The above polynomial can be changed to the following form for convenience in evaluation and for diminishing the effect of round-off error in repeated calculations. The steps in the calculation of the example are shown below.

$$x(C_1 + C_3x^2 + C_5x^4 + C_7x^6)$$

The quantity within the parentheses can be rewritten:

$$C_1 + x(C_3x + C_5x^3 + C_7x^5)$$

Again, rewriting the parenthetical quantity,

$$x(C_3 + C_5x^2 + C_7x^4)$$

and, following the same procedure for the quantity in the parentheses,

$$C_3 + x(C_5x + C_7x^3)$$

and again for the quantity within the parentheses,

$$x(C_5 + C_7x^2)$$

and finally for the quantity in parentheses,

$$C_5 + x(C_7x)$$

These can then be reassembled as follows:

$$\sin\left(\frac{\pi}{2}\cdot x\right) = x[x \mid x \langle x(x\{x[x(C_7)] + C_5\}) + C_3\rangle \mid + C_1]$$

Step	$x = 1$ (90°)	$x = 0.5$ (45°)
1. $C_7 \cdot x$	−0.004362476	−0.002181238
2. $\cdot x$	−0.004362476	−0.001090619
3. $+C_5$	0.075125187	0.078397044
4. $\cdot x$	0.075125187	0.039198522
5. $\cdot x$	0.075125187	0.019599261
6. $+C_3$	−0.570795791	−0.626321717
7. $\cdot x$	−0.570795791	−0.313160858
8. $\cdot x$	−0.570795791	−0.156580429
9. $+C_1$	0.999999061	1.414214423
10. $\cdot x$	0.999999061	0.707107211

The values listed for Step 10 are good approximations of the sine of 90 degrees and 45 degrees, respectively.

In trigonometric applications there is often need to consider inverse relationship; that is, having computed the value of an expression, determine the corresponding angle. The inverse relationships such as the arc sine and arc tangent, symbolically indicated by $\sin^{-1}$ and $\tan^{-1}$, can be computed by the series methods referred to above or by approximating the relation by a polynomial. The polynomial given as an example is taken from Hastings' book. Like the one given for the sine above, the following expression introduces an error. However, if the application warrants greater accuracy, other expressions, but with additional terms, may be obtained from the same reference:

$$\sin^{-1} x = \frac{\pi}{2} - \sqrt{1-x}(F(x))$$

where

$$F(x) = a_0 + a_1x + a_2x^2 + a_3x^3 + a_4x^4$$

and where

$$\begin{aligned} a_0 &= 1.57078786 \\ a_1 &= -0.21412453 \\ a_2 &= 0.08466649 \\ a_3 &= -0.03575663 \\ a_4 &= 0.00864884 \end{aligned}$$

The use of this method is shown below. Suppose that in a certain mathematical application the following equation appears and values of y and z are known:

$$\sin a = \cos y \sin z + \sin y \cos z.$$

It would not be difficult to evaluate the right-hand side of the equation. For this illustration, suppose that it came out to be 0.36281743. This means that

$$\sin a = 0.36281743.$$

What, though, is the measure of the *angle* the sine of which is 0.36281743? By letting $a = x$, above, and multiplying this value by $\sqrt{1 - x}$ and then subtracting this value from $\frac{\pi}{2}$, we will obtain the value, in radians, of the angle the sine of which is equal to 0.36281743.

The calculations below show that $\sin^{-1} 0.36281743 = 0.3712957$ radians, or 21.2736 degrees.

$$\begin{aligned} F(x) &= 1.57078786 - 0.21412453 \times 0.36281743 + \\ &\quad 0.08466649(0.36281743)^2 - 0.03575663(0.36281743)^3 + \\ &\quad 0.00864884(0.36281743)^4 \\ &= 1.50268708 \end{aligned}$$

$$\begin{aligned} \sqrt{1 - x} &= \sqrt{1 - 0.36281743} = \sqrt{0.63718257} \\ &= 0.79823716 \end{aligned}$$

$$\frac{\pi}{2} = \frac{3.1415927}{2} = 1.5707963$$

$$\begin{aligned} \sin^{-1} 0.36281743 &= 1.5707963 - (1.50268708 \times 0.79823716) \\ &= 1.5707963 - 1.1995006 \\ &= 0.3712957 \end{aligned}$$

Since

$$1 \text{ radian} = 57.29577^\circ,$$

$$0.3712957 \text{ radians} = 0.3712957 \times 57.29577^\circ, \text{ or } 21.2736^\circ.$$

According to Peters' *Seven Place Values of Trigonometric Functions,*

$$\sin 21.273^\circ = 0.3628121$$

and

$$\sin 21.274^\circ = 0.3628284$$

showing that the value as computed is in agreement to at least two decimal places.

Note that if the value of $\cos x$ were also needed, it could be obtained merely by rearranging the expression

$$\sin^2 x + \cos^2 x = 1$$

or

$$\cos x = \sqrt{1 - (0.36281743)^2}.$$

There are also polynomials listed in Hastings' book for the $\tan^{-1}$, one of which is as follows:

$$\tan^{-1}x = C_1x + C_3x^3 + C_5x^5 + C_7x^7$$

where

$$C_1 = 0.9992150$$
$$C_3 = -0.3211819$$
$$C_5 = 0.1462766$$
$$C_7 = -0.0389929$$

It should be pointed out that in working with the tangent function care must be exercised since for values of the angle near $\frac{\pi}{2}$ (90 degrees), the value of the function becomes very large.

All of the above indicates that with the aid of a computer it is possible and feasible to perform all trigonometric manipulations that may be required, provided it is possible to (1) evaluate the three polynomials of the type given above for $\sin x$, $\sin^{-1}x$, and $\tan^{-1}x$, and (2) extract square root. However, these procedures may introduce difficulties, such as in the sine-cosine relationships when small angles are involved, or in tangent-cotangent procedures when the angles are large.

Still another difficulty is retaining the proper algebraic sign during the calculation processes that call for the reduction of angles to their first-quadrant equivalents. The computer-oriented methods, therefore, do not eliminate the need for planning and careful attention on the part of the problem solver.

Example 6. Simultaneous Equations

Partly because of their speed, large digital computers, introduce the possibility of using problem-solving techniques that might otherwise be difficult or undesirable because of the time involved. The following method is of the type that takes advantage of a digital computer's speed and puts into a position of secondary importance the making of complicated decisions on "What to do next?" If a person could solve a problem by one method which calls for a large number of additions, subtractions, and other arithmetic steps or by another method which calls for very little arithmetic, other things being equal, he would probably choose the latter method.

In fact, mathematicians have spent much of their time in efforts to develop powerful methods for solving large classes of individually simple problems. For example, it is possible to perform multiplication by repeated addition. However, most people do not use this method. More powerful methods, using the idea of multiplication tables, have been developed; they reduce the number of arithmetic steps needed.

The following example is a method of solving equations which is quite adaptable to a large computer. The operations are simple, but many of them are required.

PROBLEM: Solve the following equations for x and y.

$$x^2 + xy = 10 \tag{A}$$

$$2x + xy^2 = 22 \tag{B}$$

Equation A can be rewritten as follows:

$$x(x + y) = 10$$

$$x = \frac{10}{x + y}. \tag{C}$$

Also, Equation B can be rewritten as follows:

$$2x + (xy)y = 22$$

$$y = \frac{22 - 2x}{xy}. \tag{D}$$

Both Equations C and D are in unusual form since x and y appear on both sides of the equal sign. This can be stated in another way: x and y are functions of x and y combined. In other words, x is somehow represented by an inter-relationship between x and y. Likewise, y is somehow represented by an inter-relationship between x and y, but it is a different relationship. Stated in functional notation,

$$x = f_1(x,y) \tag{E}$$

$$y = f_2(x,y). \tag{F}$$

By making approximations for the values of x and y on the right side of equations E and F, it may be possible to establish an iterative procedure to improve the approximation. With the improved value a still better one may be obtained, and so on, until sufficiently good values are obtained and the process is stopped.

The determination of *sufficiently good* values depends on the requirements of the particular application. For example, if in the iterative procedure two successive approximations differ by no more than say ± 0.0001 (limit ϵ), and if no greater accuracy is required, it is not necessary to continue the process.

Because the iterative procedure refers to more than one x and y, that is, to several approximations, it is desirable to establish an identification procedure so that it is easy to keep track of the process. It is customary to use subscripts for this identification where the first approximation for x is called x_0, the second approximation for x is called x_1, the third approximation is called x_2, etc. A similar plan is used to keep track of the y's.

Using this notation then, Equation E would be rewritten for the first step of the process as follows:

$$x_1 = f_1(x_0, y_0).$$

Using the particular example above, Equation C would be written as:

$$x_1 = \frac{10}{x_0 + y_0}. \tag{G}$$

In a similar manner, Equation F for y would be written as:

$$y_1 = f_2(x_1, y_0)$$

and for this particular example, Equation D would be written as:

$$y_1 = \frac{22 - 2x_1}{x_1 y_0}. \qquad \text{(H)}$$

Notice that, in Equation H, the "new" x (that is, x_1, just obtained) is used to compute the "new" y. As soon as both the new x and new y are available, the process is repeated, using Equations G and H with the subscripts increased by 1. As the process settles down, or converges, so that, say, two successive trials within certain limits (such as a 5 in the third decimal place) are unchanged, the process can be stopped. A flow diagram for the general process is shown in Fig. 3–10.

The above example is shown in detail below using starting values of $x_0 = 1$ and $y_0 = 4$. To obtain the starting values it is possible to substitute values in the individual equations and plot the results on graph paper. By noting where the two curves thus plotted intersect, estimates suitable for starting the process can be obtained. The illustration shows an intersection at $x = 2$, $y = 3$. To show how the process converges, $x_0 = 1$ and $y_0 = 4$ were chosen. In actual practice the graphical solution, although apparently adequate in this example, might be inadequate because a graph cannot be read or drawn with the accuracy that might be needed—such as to the fourth decimal place—except by special, and rather laborious, procedures.

An instructive exercise is to plot the successive approximations that are produced by the process. This shows in a graphical way how the process converges.

The procedure for selecting starting values—Equation A, $x^2 + xy = 10$, is written as follows:

$$y = \frac{10 - x^2}{x} \qquad \text{(I)}$$

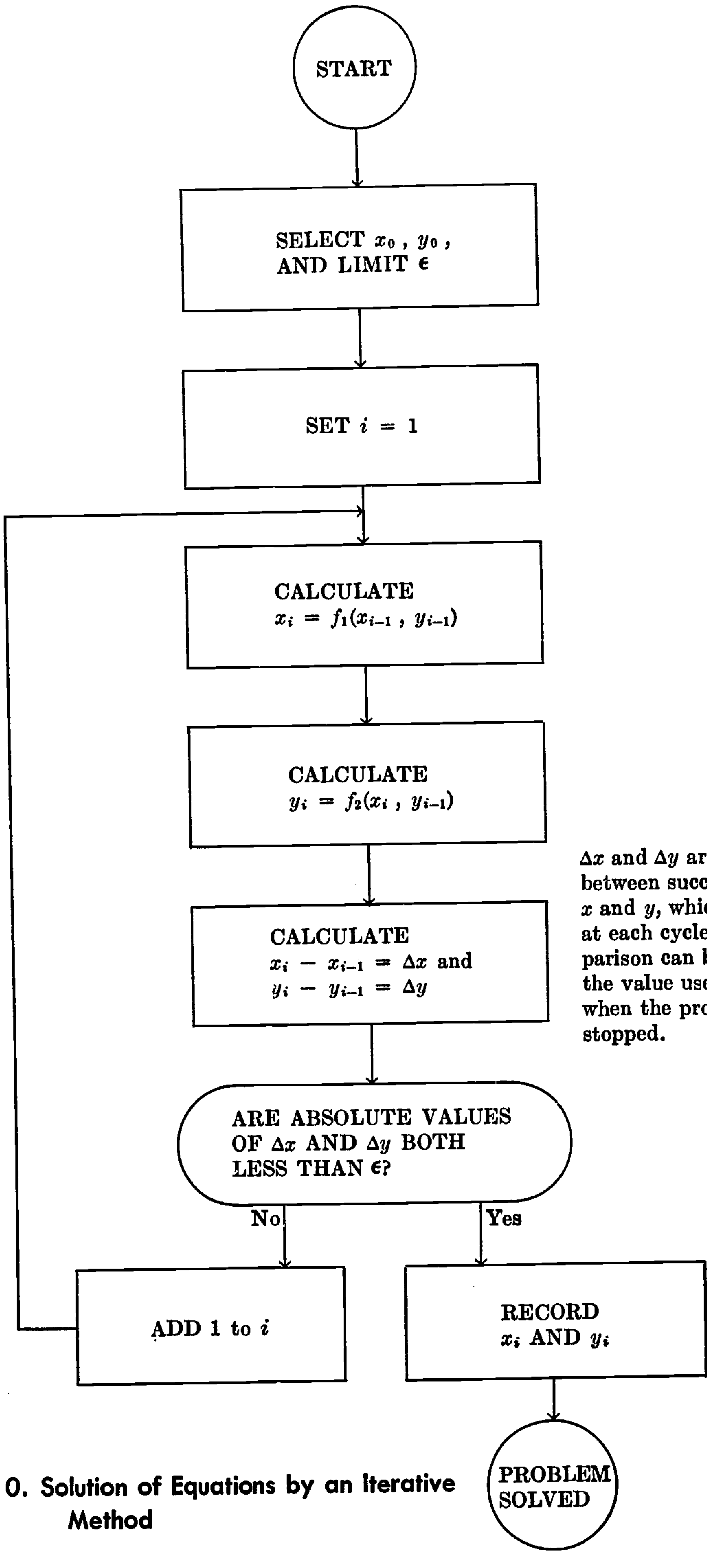
and for values of x, which are substituted, corresponding values of y are obtained. In the same way Equation B, $2x + xy^2 = 22$, is written as follows:

$$x = \frac{22}{2 + y^2}. \qquad \text{(J)}$$

In like manner, values of y are substituted to obtain corresponding values of x. These tabulations (to one place only) follow and are plotted in Fig. 3–11.

For Equation I		For Equation J	
x	y	y	x
1	9.0	1	7.3
2	3.0	2	3.6
3	0.3	3	2.0
4	−1.5	4	1.2

Although the starting values of $x_0 = 2$ and $y_0 = 3$ would be good to select on the basis of the intersection of the two curves, the values $x_0 = 1$ and $y_0 = 4$



Δx and Δy are the differences between successive values of x and y, which are determined at each cycle so that a comparison can be made with ϵ, the value used to indicate when the procedure can be stopped.

Fig. 3-10. Solution of Equations by an Iterative Method

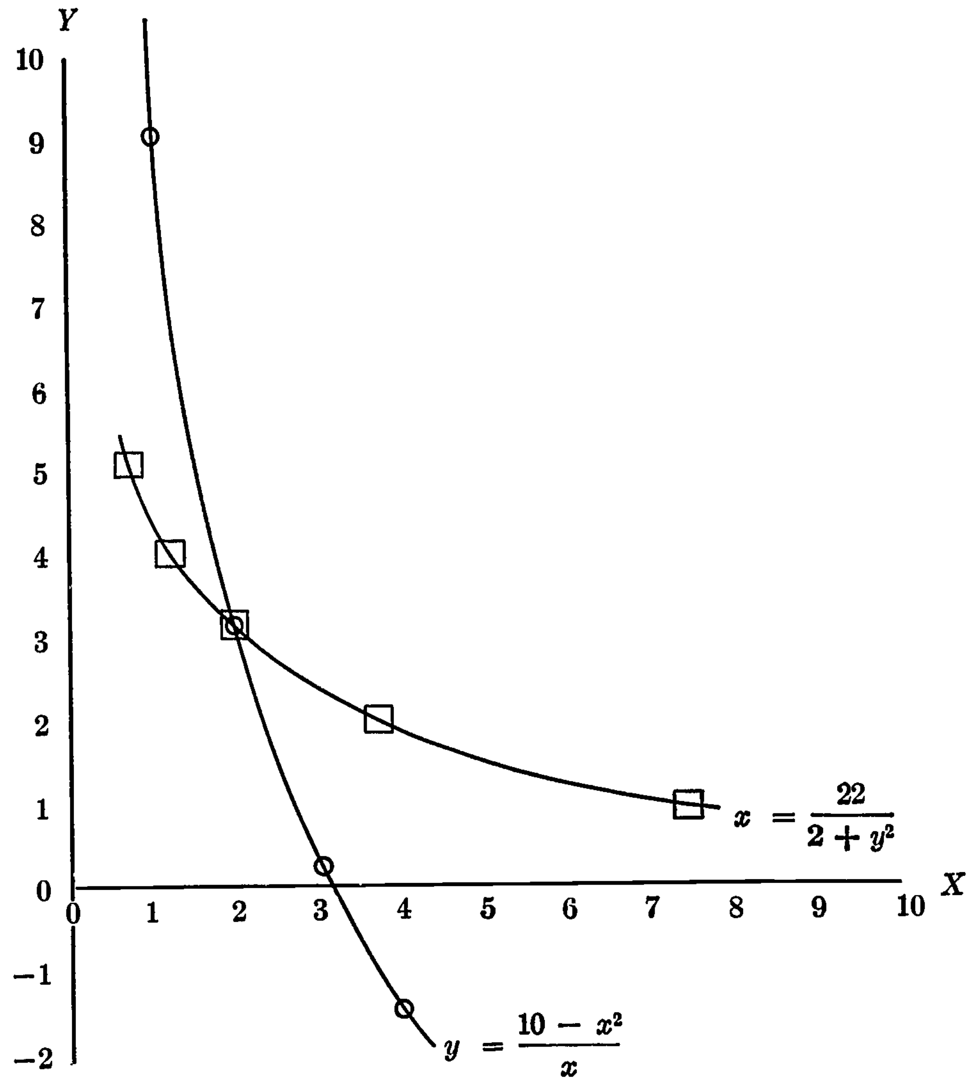


Fig. 3-11. Graph of Equations (I) and (J)

were chosen as being better for demonstration purposes. The remaining ten steps of the process are as follows:

	Δx	Δy
$x_1 = \dfrac{10}{1.00 + 4.00} = \dfrac{10}{5.00} = 2.00$	1.00	
$y_1 = \dfrac{22 - (2 \times 2.00)}{2.00 \times 4.00} = \dfrac{22 - 4}{8.00} = 2.25$		−1.75
$x_2 = \dfrac{10}{2 + 2.25} = \dfrac{10}{4.25} = 2.35$	0.35	
$y_2 = \dfrac{22 - (2 \times 2.35)}{2.35 \times 2.25} = \dfrac{22 - 4.70}{5.29} = 3.27$		1.02
$x_3 = \dfrac{10}{2.35 + 3.27} = \dfrac{10}{5.62} = 1.78$	−0.57	
$y_3 = \dfrac{22 - (2 \times 1.78)}{1.78 \times 3.27} = \dfrac{22 - 3.56}{5.82} = 3.17$		−0.10

$x_4 = \dfrac{10}{1.78 + 3.17} = \dfrac{10}{4.95} = 2.02$	0.24	
$y_4 = \dfrac{22 - (2 \times 2.02)}{2.02 \times 3.17} = \dfrac{22 - 4.04}{6.40} = 2.81$		−0.36
$x_5 = \dfrac{10}{2.02 + 2.81} = \dfrac{10}{4.83} = 2.07$	0.05	
$y_5 = \dfrac{22 - (2 \times 2.07)}{2.07 \times 2.81} = \dfrac{22 - 4.14}{5.82} = 3.07$		0.26
$x_6 = \dfrac{10}{2.07 + 3.07} = \dfrac{10}{5.14} = 1.95$	−0.12	
$y_6 = \dfrac{22 - (2 \times 1.95)}{1.95 \times 3.07} = \dfrac{22 - 3.90}{5.99} = 3.02$		−0.05
$x_7 = \dfrac{10}{1.95 + 3.02} = \dfrac{10}{4.97} = 2.01$	0.06	
$y_7 = \dfrac{22 - (2 \times 2.01)}{2.01 \times 3.02} = \dfrac{22 - 4.02}{6.07} = 2.96$		−0.06
$x_8 = \dfrac{10}{2.01 + 2.96} = \dfrac{10}{4.97} = 2.01$	0.00	
$y_8 = \dfrac{22 - (2 \times 2.01)}{2.01 \times 2.96} = \dfrac{22 - 4.02}{5.95} = 3.02$		0.06
$x_9 = \dfrac{10}{2.01 + 3.02} = \dfrac{10}{5.03} = 1.99$	−0.02	
$y_9 = \dfrac{22 - (2 \times 1.99)}{1.99 \times 3.02} = \dfrac{22 - 3.98}{6.01} = 3.00$		−0.02
$x_{10} = \dfrac{10}{1.99 + 3.00} = \dfrac{10}{4.99} = 2.00$	0.01	
$y_{10} = \dfrac{22 - (2 \times 2.00)}{2.00 \times 3.00} = \dfrac{22 - 4.00}{6.00} = 3.00$		0.00

This series is shown graphically in Fig. 3–12.

It should be pointed out that the above example is merely intended to demonstrate a possible procedure and to call attention to iterative methods. This method may run into practical difficulties if the solutions are such that machine capacity is exceeded, as, for instance, when x is very small and y very large. Many questions can develop in the analysis of such methods, one of the most important of which is: How can one be certain that the procedure will finally converge? This question cannot be answered here except to the extent that in some circumstances convergence will not occur and in other circumstances the convergence may be so slow that the method is impractical. The study of convergence is an important part of numerical analysis and is treated in various textbooks.

Briefly, the situation can be represented by the diagrams below, where x repre-

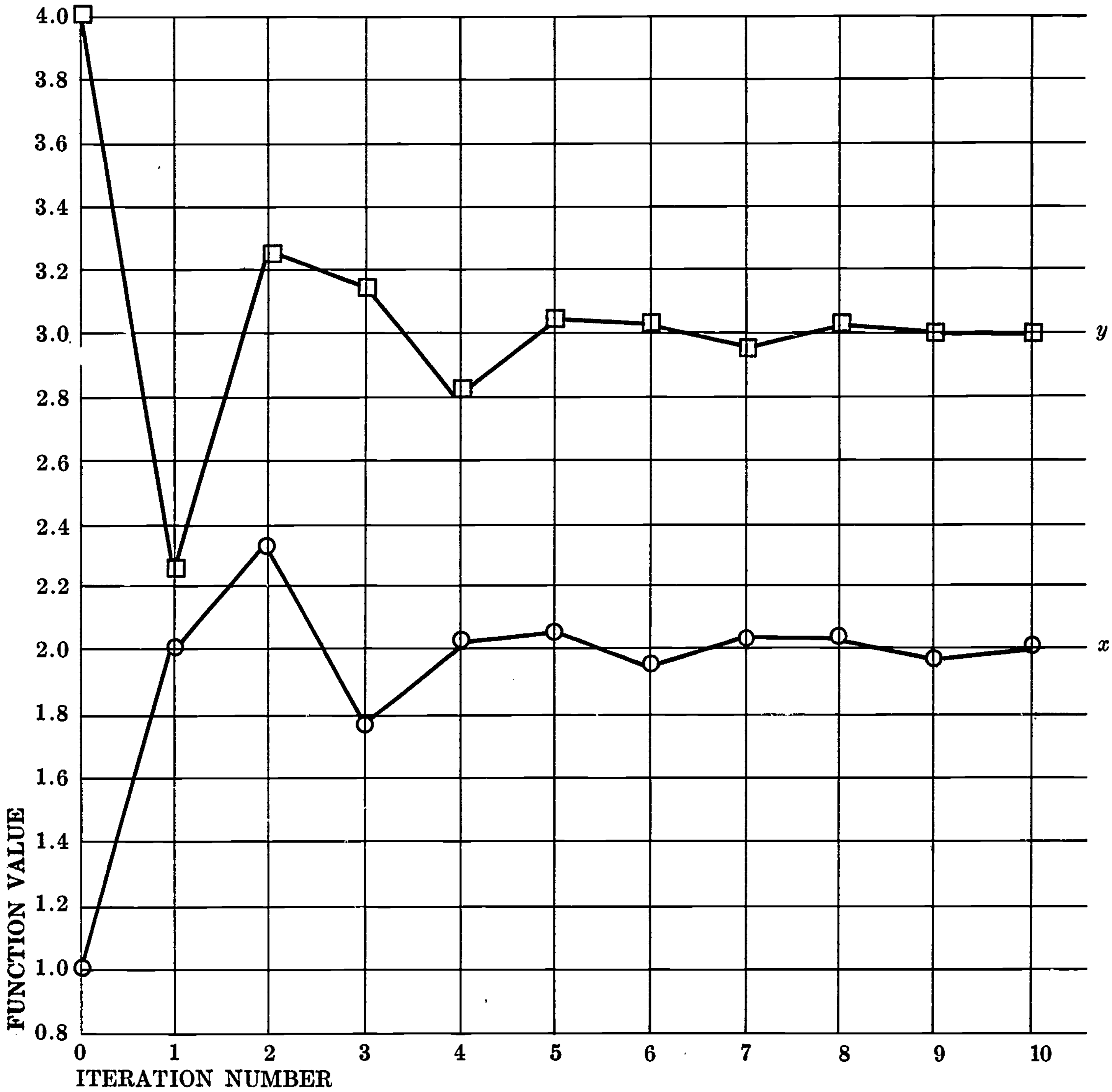
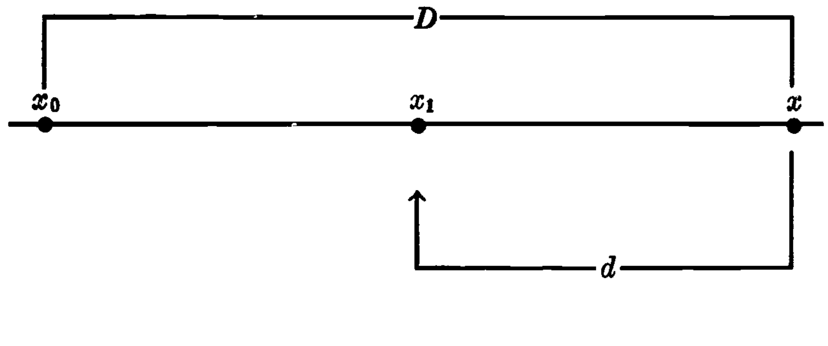
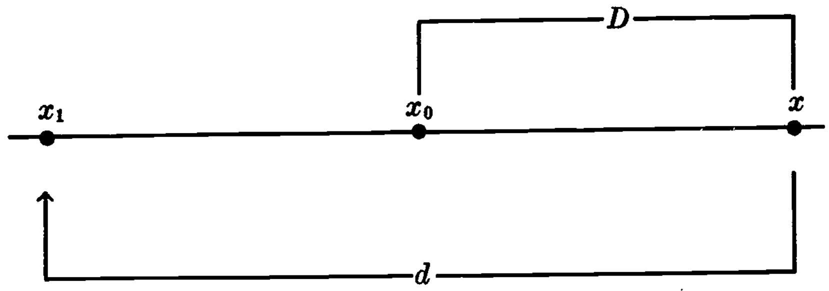


Fig. 3-12. An Iterative Process Settling Down

sents the value being sought, x_0 the first approximation, and x_1 the value produced by using x_0 in the iterative procedure. When these are represented as points on a line, such as the one below, it can be seen that the iterative procedure was such that an improvement was made.



That is, the original difference, D (where $D = x - x_0$), was reduced to d (where $d = x - x_1$). The opposite might have happened:



Merely "going in the wrong direction" does not mean that the process will not produce the desired result, since, as shown in Fig. 3–13, a procedure may approach the solution from one side or it may oscillate around it. To show the effect of successive trials, the x's are connected by lines to show how processes may or may not converge.

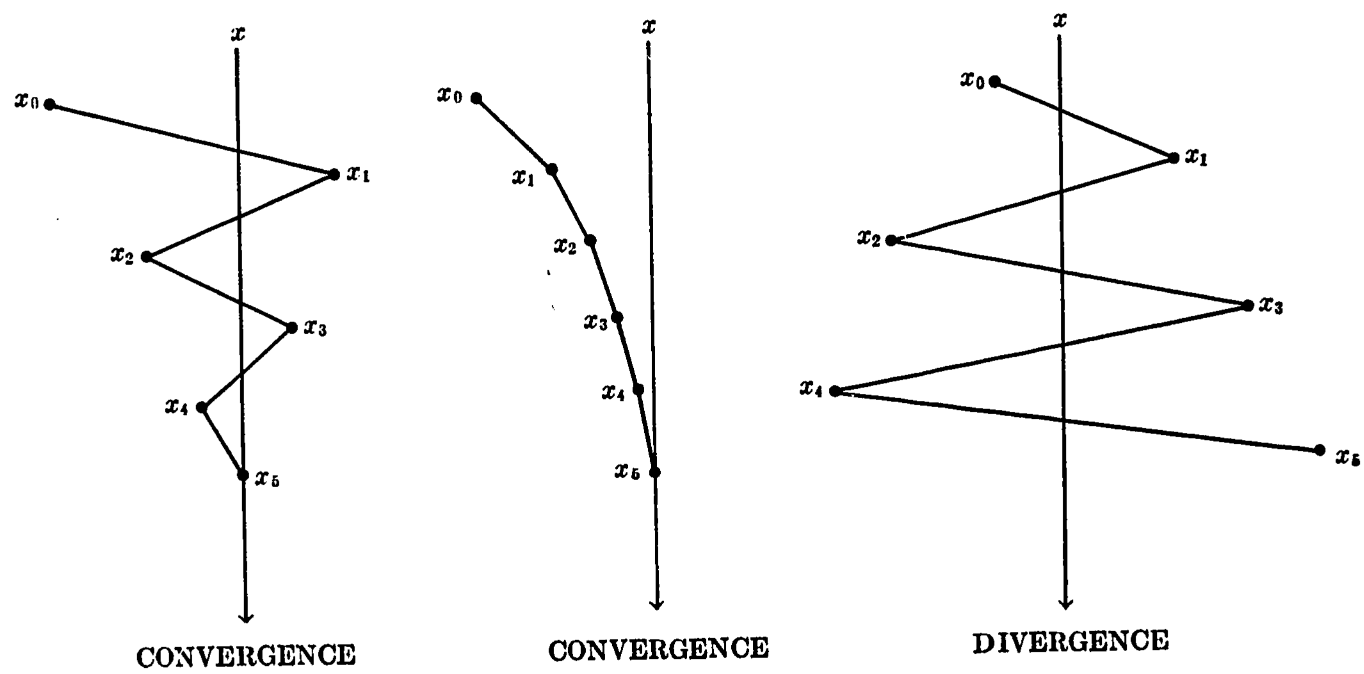


Fig. 3-13. Types of Convergence and Divergence

Example 7. Computations Involving Rate of Change

Part A

One of the most important concepts in mathematics pertains to operations performed on relationships that are not static but dynamic; that is, the treatment of expressions that involve changing situations. This idea is most important because in the everyday world things are dynamic rather than static. In studying either the motion of a rocket, the heating of a kitchen stove, or the growth of living organisms, the concept of change must be considered. For example, it is well known that if the speed of an automobile is given as 60 miles per hour and the total travel time is given as 10 hours, the total distance traveled is 600 miles. There is, in fact, the following relationship that applies to such problems:

$$d = rt$$

where $d =$ the measure of the distance, $r =$ the measure of the rate of travel and

t = the measure of the time traveled. However, there are few cases in real life in which such a situation exists. In most cases the actual rate is zero to start with, and it increases and decreases while the trip is under way. If the variations in speed are such that the *average* rate is equal to 60, then the above expression is true. The same argument would apply to other situations in nature . . . as time elapses, whether it is while a car is being driven, a plant is growing, or a rocket is moving, the parameters describing the situation quite often change in value. The following example will indicate some of the problems involved: As a rocket leaves the launching pad, many factors influence its trajectory; some of these are listed below.

1. The weight of the rocket
2. The amount of thrust delivered by the rocket motor
3. The density of the air through which the rocket must travel

The weight of course, is not constant, since as fuel is burned the whole rocket becomes increasingly lighter until all the fuel is consumed. In other words, the weight is not a constant value—it changes continually.

The effect of the thrust—even if the thrust itself is held to a constant value—may not be constant if only because the whole rocket is getting lighter as fuel is burned. Therefore, thrust of a certain value may push the rocket more effectively during the third stage of flight than during the second, since the rocket is lighter.

The density of the atmosphere is also a variable that affects the rocket's flight, since as one goes higher and higher above the earth, there is less and less atmosphere to offer resistance. The density of the atmosphere, however, does not change in a simple fashion—it is much more dense near the earth, and about half of the entire atmosphere is below 18,000 feet. The *rate* at which the density is changing is *itself* changing. But, back to the rocket, thrust of a certain value will have a different effect on the rocket depending on the air density at that time; however, "at that time" is somewhat meaningless, since as the rocket moves, the altitude changes and so the density changes.

Thus it can be seen that if factors such as thrust, density, and weight become interrelated while being subject to considerable variation, the problem of describing the result of their relationship (for example: How far has the rocket gone at the end of ten seconds of flight?) would be very nearly impossible without some means of handling, mathematically, quantities that are in the process of change.

One way to consider the way in which quantities change is to imagine that instantaneous observations of the process can be made and recorded by some means. For example, think of an automobile starting on a trip. Its velocity might be recorded at various times as follows:

Time, seconds	Velocity, mph	Difference
0	0	
		2
1	2	
		2
2	4	
		2

Time, seconds	Velocity, mph	Difference
3	6	
		2
4	8	
		2
5	10	
		2
6	12	
		2
7	14	
		2
8	16	
		2
9	18	

The values in this table mean that at the instant the second hand on the stop watch shows 4 seconds, the speedometer shows 8 miles per hour, and that at the instant the stop watch shows 8 seconds the speedometer shows 16 miles per hour, and so forth. Understand that these were not constant velocities, merely values of velocity "caught on the fly."

The inadequacy of the formula $d = rt$ under these circumstances can be understood by attempting to determine how far a car would have traveled by the time the second hand on the stop watch shows, say, 8 seconds.

Data of the above type can be examined by obtaining differences of adjacent values as shown by the third column in the table. Notice that the differences in this example are all equal to 2. This indicates that the velocity, which goes from 0 to 18, is changing at a constant rate; that is, at the end of each second the velocity is 2 miles per hour more than it was at the end of the previous second. This change in the velocity is called *acceleration*. In like manner, velocity can be considered *a change in distance*.

The rate of change in velocity (acceleration) need not be constant, of course. The following table will demonstrate this.

Time	Velocity	Acceleration
0	0	
		2
1	2	
		4
2	6	
		6
3	12	
		8
4	20	
		10
5	30	
		12
6	42	
		14
7	56	
		16
8	72	
		18
9	90	

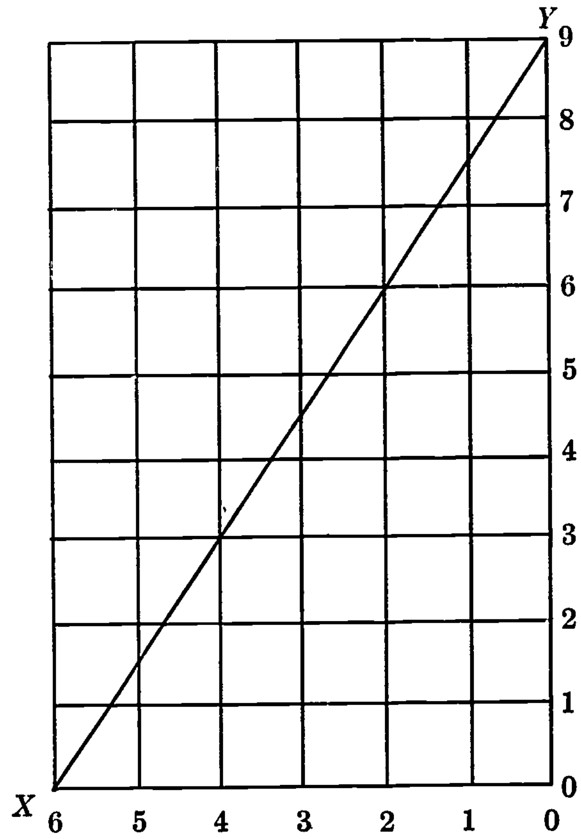


Fig. 3-14. A Triangular Area To Be Measured

The important idea here is that physical situations, as observed in nature, often vary with respect to the time during which they occur. It also frequently happens that acceleration can be observed and measured more readily than velocity. However, from such measurements it is often necessary to obtain the velocity. It is, therefore, desirable to be able to operate with these varying quantities in such a way as to "know where they come from." For example, in the tables given above it would be possible, say, given the acceleration, to construct the column of velocities. (It has been already shown that, given the velocity, the acceleration, as represented by the differences, can be obtained.)

Part B

For solving problems associated with changing quantities there are methods that depend on operations performed upon the successive differences, or sums, of the quantities in question. The following example shows one such method. It is applied to the problem of finding the area of a triangle, making use of several values of x, in the graph, and the corresponding values of y.

Suppose it is desired to obtain the area of the triangle shown in Fig. 3–14. (Note that the coordinate numbers increase from *right to left.*)

The following table lists x and y as shown in the graph. For example, for $x = 1$, $y = 7.5$, etc. Successive differences of x, successive sums of y, and their products are also listed.

x	Difference	y	Sum	Product of difference and sum
0		9.0		
	1		16.5	16.5
1		7.5		
	1		13.5	13.5
2		6.0		
	1		10.5	10.5
3		4.5		
	1		7.5	7.5
4		3.0		
	1		4.5	4.5
5		1.5		
	1		1.5	1.5
6		0.0		
				54.0 ÷ 2 = 27.0

This is actually a representation of how one quantity is changing as another also changes. That is, for different values of x, y also has different values. By multiplying the differences by their corresponding sums and then taking one half of the sum of the products, the area is obtained. This, of course, can be done "the other way" as shown below:

x	Sum	y	Difference	Product of difference and sum
0		9.0		
	1		1.5	1.5
1		7.5		
	3		1.5	4.5
2		6.0		
	5		1.5	7.5
3		4.5		
	7		1.5	10.5
4		3.0		
	9		1.5	13.5
5		1.5		
	11		1.5	16.5
6		0.0		
				54.0 ÷ 2 = 27.0

An examination of this procedure indicates that what is happening is that the triangle is treated, a piece at a time, as a succession of rectangles, as shown in Fig. 3–15.

To gain some understanding of the above procedure, suppose the formula for finding the area of a rectangle is known. Then it would be possible to use a method like the one which follows to determine the area of the triangle.

The area of the lower rectangle is 1.0 by 7.5 and the area of the entire longer rectangle is 1.0 by 9.0. A good estimate for the area of the part of the triangle between $x = 0$ and $x = 1$ is the average of the areas of the larger and smaller rectangles, mentioned above; that is:

$$(\tfrac{1}{2})[(1.0 \times 9.0) + (1.0 \times 7.5)] \text{ or } (\tfrac{1}{2})(9.0 + 7.5).$$

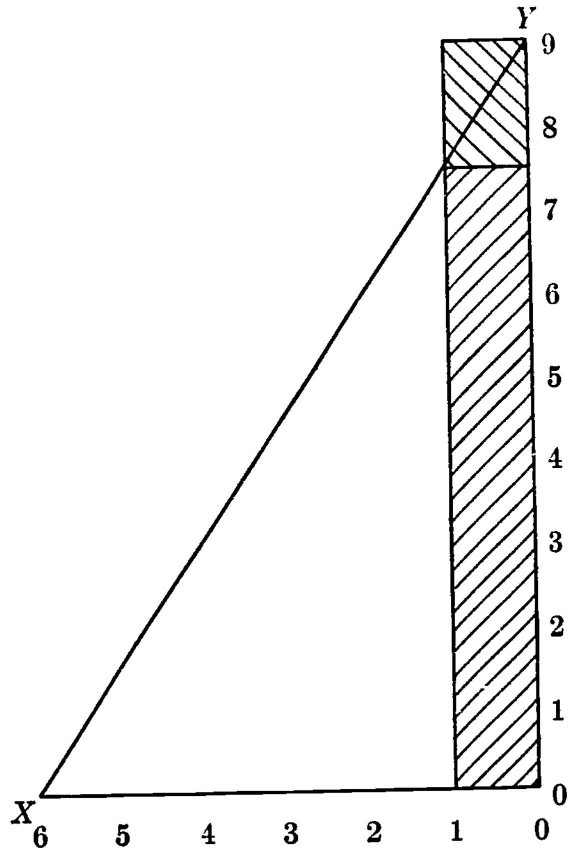


Fig. 3-15. Measuring a Triangular Area by Successive Rectangles

In other words, the smaller rectangle is too small, since part of the triangle is not included; also the larger rectangle is too large, since it includes some area outside the triangle. The average of the two rectangles is better than either rectangle, taken by itself. To find the area of the whole triangle one would proceed across the triangle, taking the average of successive rectangles and finally obtaining the sum of the average values.

Notice that if the values of x are equally spaced, as in the example, the above procedure can be simplified. Instead of obtaining the successive subtractions for each value of x, it is necessary to do so only once, since all differences are the same. It is customary to refer to this interval by the letter h.

As just shown, the area of the first slice of the triangle would be:

$$\frac{h}{2}(9.0 + 7.5).$$

The remaining slices would be as follows:

$$\frac{h}{2}(7.5 + 6.0)$$

$$\frac{h}{2}(6.0 + 4.5)$$

$$\frac{h}{2}(4.5 + 3.0)$$

$$\frac{h}{2}(3.0 + 1.5)$$

$$\frac{h}{2}(1.5 + 0.0).$$

Notice that each value of y, except the first (9.0) and the last (0.0), appears twice. This means that the value 7.5, for example, is multiplied by $\frac{h}{2}$ in the first expression as well as in the second. Since the several slices have to be added together to get the final area, it means that the 7.5 is divided by 2 (first expression), later on (second expression) independently divided by 2, and the results of these two divisions are subsequently added together. Such divisions can, of course, then be avoided, except for the first and last values.

The sum of the above expressions equals the area of the triangle and is equivalent to the more simplified expression below:

$$\text{Area} = h\left(\frac{9.0}{2} + 7.5 + 6.0 + 4.5 + 3.0 + 1.5 + \frac{0.0}{2}\right).$$

This is known as the trapezoidal rule for numerical integration, since it effectively divides the area into a group of trapezoids which are then added together. It is noted for its simplicity, since for equally spaced values of one variable, x, the corresponding values of the other variable, y, are merely added together (except for the first and last values which are divided by 2 before adding), and the sum is multiplied by the value corresponding to the size of the interval. The principal disadvantage of the method is that for certain types of functions, that is, those that are not represented by a straight line, as in the above example, accuracy may not be high.

Part C

A method which is superior to the trapezoidal rule because it provides higher accuracy is the one known as Simpson's rule. It is given below for the above example, without any development:

$$\text{Area} = \frac{h}{3}(9.0 + 4(7.5) + 2(6.0) + 4(4.5) + 2(3.0) + 4(1.5) + 0.0).$$

This method is a simple one, since for equally spaced intervals all that is required is (1) to obtain the sum of the even-numbered terms (7.5, 4.5, 1.5) and multiply by 4, (2) to obtain the sum of the odd-numbered terms except the first and last (6.0, 3.0), and multiply by 2, (3) obtain the sum of the first and last terms, and (4) multiply the sum of these three values by $h/3$. The principal disadvantage of this method is that an even number of intervals is required.

Both methods are widely used in numerical calculations. They consist of simple repetitive processes that are adaptable to calculation, either with the use of a desk calculator or a large digital computer.

It is not likely that the area of a triangle would be computed by methods such as shown above, but these methods can be applied to other figures the sides of which are not straight lines.

With a piece of graph paper it is possible to draw a circle and determine its area by these procedures, although it may be discovered that the area thus determined will not agree exactly with the area determined by the formula $A = \pi r^2$. One reason for the lack of agreement is that the measurements cannot be made accurately enough. However, if the circle is drawn on very large-scale graph paper with very fine divisions on it, the accuracy can be improved over that obtained by using a smaller figure and coarse divisions.

All of the above suggests, then, that methods exist for determining the area of a figure the sides or boundaries of which are expressed by the relationship between two quantities. For example, if corresponding values of x and y were given for Fig. 3–16, its area could be determined by the above procedures. The *rectangle-averaging* process is demonstrated.

The area of the small rectangle is $(x_2 - x_1)y_1$. The area of the larger rectangle is $(x_2 - x_1)y_2$. The area between the curve $f(x)$ and the x axis, bounded by the lines $x = x_1$ and $x = x_2$, lies between the areas of the two rectangles above; that is, the area is greater than the lower rectangle with the side x_1, y_1 but smaller than the rectangle with the side x_2, y_2. Again, the average of the two areas is an approximation of the desired area. The average can be expressed as:

$$(\tfrac{1}{2})[(x_2 - x_1)y_1 + (x_2 - x_1)y_2] \text{ or } (\tfrac{1}{2})(x_2 - x_1)(y_2 + y_1)$$

which is essentially the process of finding one-half of the product of differences and sums.

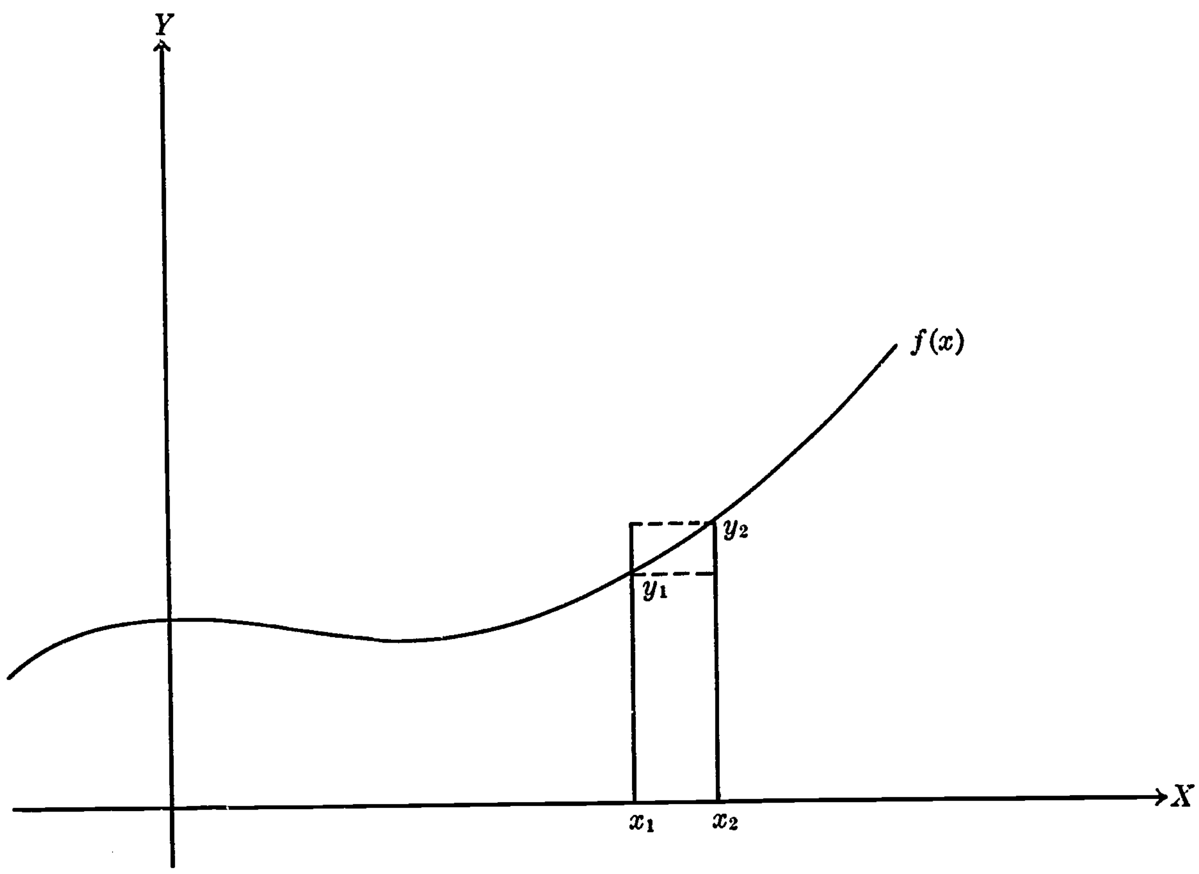


Fig. 3-16. Measuring the Area under a Curve

The error in this process is related to the change in y over a given interval in x; that is, as the line connecting y_1 and y_2 in Fig. 3–16 takes on greater and greater curvature, the error introduced by the use of a rectangular strip of a given width in the manner described above increases. Apparently then, if a figure can be represented, perhaps on graph paper, and it is possible to determine somehow a value of y for certain values of x, the area of the figure—within certain limits of error—can be determined. It is often possible to carry out this process without actually plotting it on paper. The possibility depends on whether the curve can be expressed in the form of an equation. If this can be done, the need to plot the curve and make measurements is eliminated, since the several values of y can be obtained for the corresponding values of x by numerical processes. Likewise, the processes of differencing and summing, as indicated above, can also be performed by repetitive machine processes. For example, consider the relationship expressed by the equation:

$$y = x^2 + 2$$

The relationship can be expressed by filling in the following table of values:

x	y
0	2
1	3
2	6
3	11
4	18
5	27
6	38

The curve, when plotted, looks like Fig. 3–17.

To obtain greater and greater accuracy in finding the area bounded by the curve, more values of x may be substituted and the corresponding values of y obtained. For example, using the above equation, the region between $x = 5$ and $x = 6$ can be broken down into 100 sections using:

x	y
5.00	27.00
5.01	27.10
5.02	27.20
5.03	27.30
etc.	etc.

The errors introduced by the averaging process are thus diminished since the line corresponding to the $y_1\ y_2$ part of the curve comes closer and closer to being a *straight* line. If it were perfectly straight, the error would be zero. In general, the sharper the curve—that is, the greater its curvature—the more values are needed to preserve accuracy.

An undesirable aspect of such numerical methods is that they call for so much arithmetic. However, when a computer is used, the added arithmetic burden is not so important.

Part D

In the early paragraphs of this example, it was pointed out that in certain physical problems it is important to be able to convert values of acceleration to

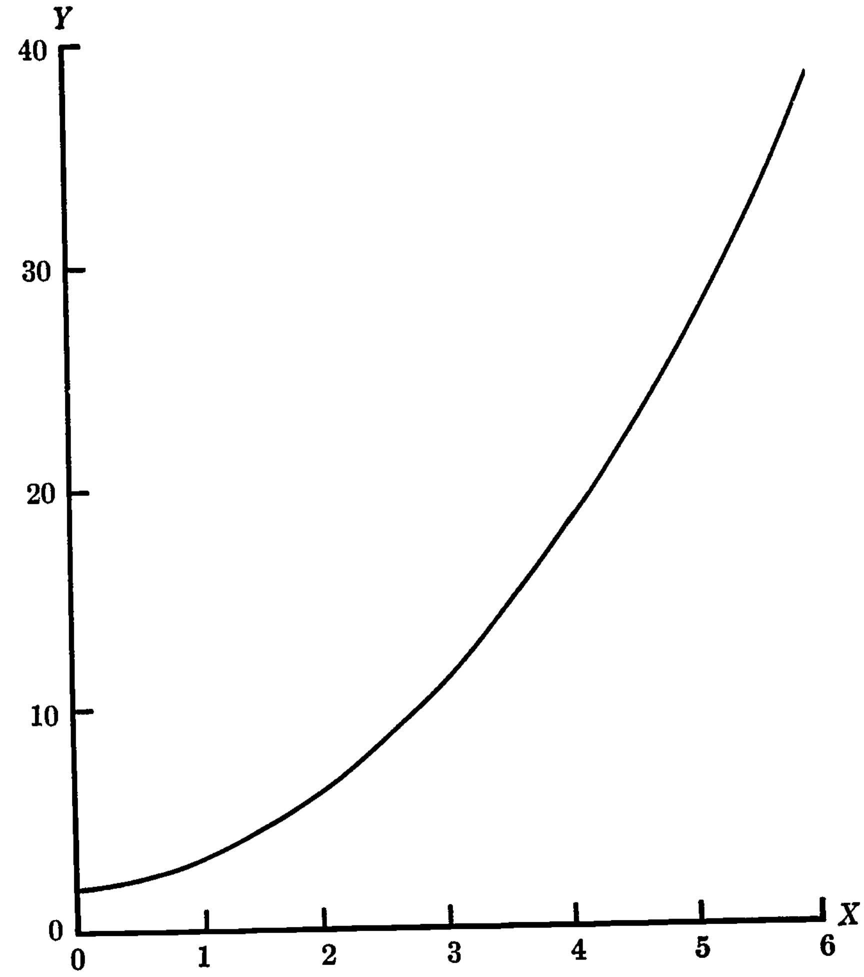


Fig. 3-17. Graph of $y = x^2 + 2$

corresponding values of velocity. The conversion is accomplished by the mathematical process of integration. Likewise, the areas of figures such as those shown above are also determined by a process of integration. In both cases—either (1) in determining velocity, given acceleration or (2) in determining area, given a mathematically described figure—the process can be of the same type.

The tabulation below shows how the sum-difference method, used above to obtain area, can be applied to the column of accelerations in the earlier example to produce a column of velocity values. The values of time and acceleration in Table 3–3 are reproduced from the table in Part A of this example. Column T is obtained by determing the value of time associated with a particular acceleration. In the differencing process, the acceleration values become *offset*; they correspond to a value of time *in between* the values recorded for the velocities. In other words, in the example given below, if the velocities are recorded at 10 and 11 seconds, the difference (acceleration) corresponds to a time somewhere between 10 and 11 seconds.

Time	Velocity	Acceleration
10	32	
		6
11	38	

TABLE 3-3

Time	Acceleration	Acceleration differences	T (explained below)	T (sums)	One half of (sum × difference)	Cummulative value of last column equals velocity
0						
	2		0.5			
1		2		2	2	2
	4		1.5			
2		2		4	4	6
	6		2.5			
3		2		6	6	12
	8		3.5			
4		2		8	8	20
	10		4.5			
5		2		10	10	30
	12		5.5			
6		2		12	12	42
	14		6.5			
7		2		14	14	56
	16		7.5			
8		2		16	16	72
	18		8.5			
9						

The value 1.5 in the column headed T in Table 3–3 means that the acceleration value of 4 was some time after one second and some time before two seconds. The 1.5 is merely an estimate based on the assumption that the situation is linear; that is, that the y_1y_2 line referred to above is straight, not curved. If it were not linear, error would be introduced into the integration process. As the time interval might be reduced, the error would also be reduced. This is the same as the suggestion above (the example using equation $y = x^2 + 2$), where the number of values in a particular region was increased.

The above processes merely suggest that integration can be performed numerically by using successive values of the function that needs to be integrated. As pointed out before, Simpson's rule is superior to the sum-difference method first described, but is similar in that successive values of the function are used. Of course, if the function can be easily integrated analytically, there may not be sufficient reason to do the work numerically.

Very often, however, the function itself is either not available or is too complicated to be solved analytically. This is the case in certain types of missile guidance, for instance, where the output of an electronic component of the guidance system is the value of the acceleration. In effect, this component produces the equivalent of a series of values (numbers), say, one each 0.01 second. The information can then be integrated—in a manner somewhat like the methods shown above—and corresponding values of velocity can be obtained. By further integrating the values of velocity, again using similar methods, values of distance can be obtained. This can be diagrammed as follows:

As a car moves along the road
its *distance* from the starting
point is *changing.*

The change in distance
is called *velocity.* If the
velocity is also *changing,*
it is called

Acceleration

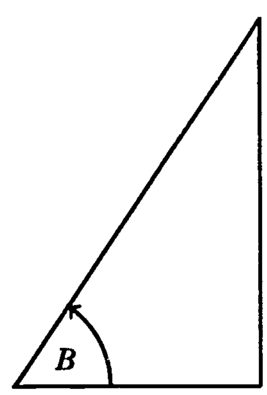
and, if one has acceleration, it is possible to work backwards:

Second integration gives *distance* ← First integration gives *velocity* ← Start

This principle, of course, is not applied exclusively to distances, velocities, and accelerations. It applies also to other situations in which the variables that are interrelated are subject to change.

The paragraphs below show a process of double integration using the trapezoidal rule.

It is helpful to consider the geometric relationship between variables as they change with respect to time.

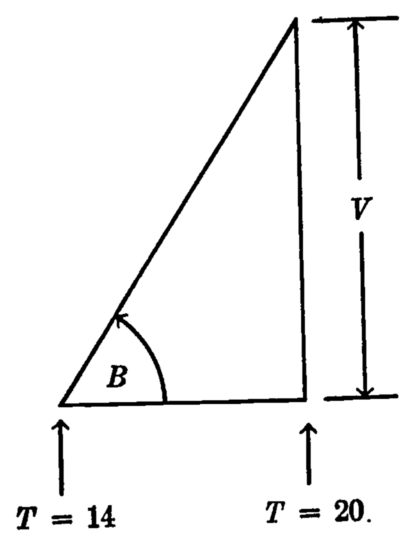


Using this representation, the slope of the hypotenuse of the triangle corresponds to the rate of change of the *variable* (called acceleration in the case of a variable velocity). The rate of change is equal to the tangent of angle B. The fact that the hypotenuse is a straight line indicates that over the period of time indicated the acceleration (rate of change of the velocity) is constant.

Suppose in the case of the missile guidance system mentioned above, the acceleration is found to be 32 over several seconds. This could be tabulated as follows:

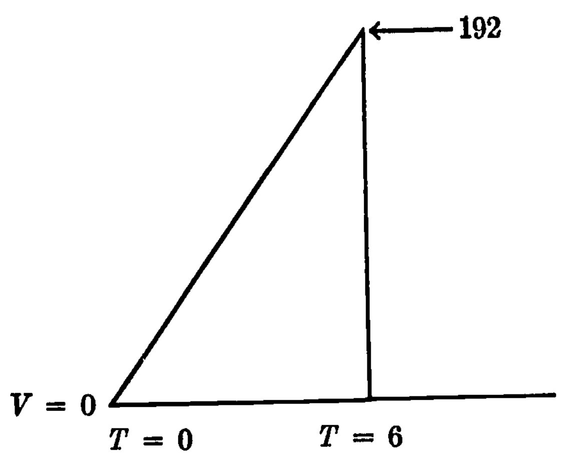
Time	Acceleration
14	32
15	32
16	32
17	32
18	32
19	32
20	32

or geometrically,



in which the tangent of angle $B = 32$.

If $\tan B = 32$ and the time increases from 14 to 20, or a total of 6, V must be 192, since $192/6 = 32$. This means that whatever the velocity was at time = 14, it was greater than that value by 192 when time = 20. There is something missing, however, if starting at a given time, one desires (1) to integrate the acceleration to obtain the actual velocity and (2) further to integrate the velocity to obtain the actual distance. The missing element is the knowledge of the situation prior to the time of interest. (For instance, suppose it is known that during a trip the automobile's average velocity increases by 4 miles per hour during each minute of travel, over a period of six minutes. However, this information by itself is sufficient neither to determine the actual velocity of the automobile nor the total distance travelled. It is known only that the velocity and distance are increasing. Another way of looking at this is to consider the problem of finding the area under a curve the boundaries of which are not known—that is, it is not known where the curve begins and ends.) For example, assume that the situation starts at zero.



This is the same geometric relationship as the one above, except that the starting values are now given; that is, it is known that at $T = 0$, $V = 0$. Therefore, knowing the acceleration, the time, and the starting values for velocity (and therefore distance), the following tabulation can be made.

Time	Distance	Velocity	Acceleration
0	0	0	0
1			32
2			32
3			32
4			32
5			32
6			32
7			

However, the above triangle might have occurred in either of the situations shown in Fig. 3–18. In other words, without knowing the starting values (that is, past history), it is not possible to proceed from a given point in time to obtain definite values for velocity and distance.

By applying the trapezoidal rule and using the hypothetical accelerations and starting values given above, the following values for velocity are obtained, by letting $h = 1$ and integrating the values for acceleration.

Integration 1. Time = 0 to Time = 1:

$$\text{Velocity} = 1\left(\frac{0}{2} + \frac{32}{2}\right) = 16$$

Integration 2. Time = 0 to Time = 2:

$$\text{Velocity} = 1\left(\frac{0}{2} + 32 + \frac{32}{2}\right) = 48$$

Integration 3. Time = 0 to Time = 3:

$$\text{Velocity} = 1\left(\frac{0}{2} + 32 + 32 + \frac{32}{2}\right) = 80$$

Integration 4. Time = 0 to Time = 4:

$$\text{Velocity} = 1\left(\frac{0}{2} + 32 + 32 + 32 + \frac{32}{2}\right) = 112$$

Integration 5. Time = 0 to Time = 5:

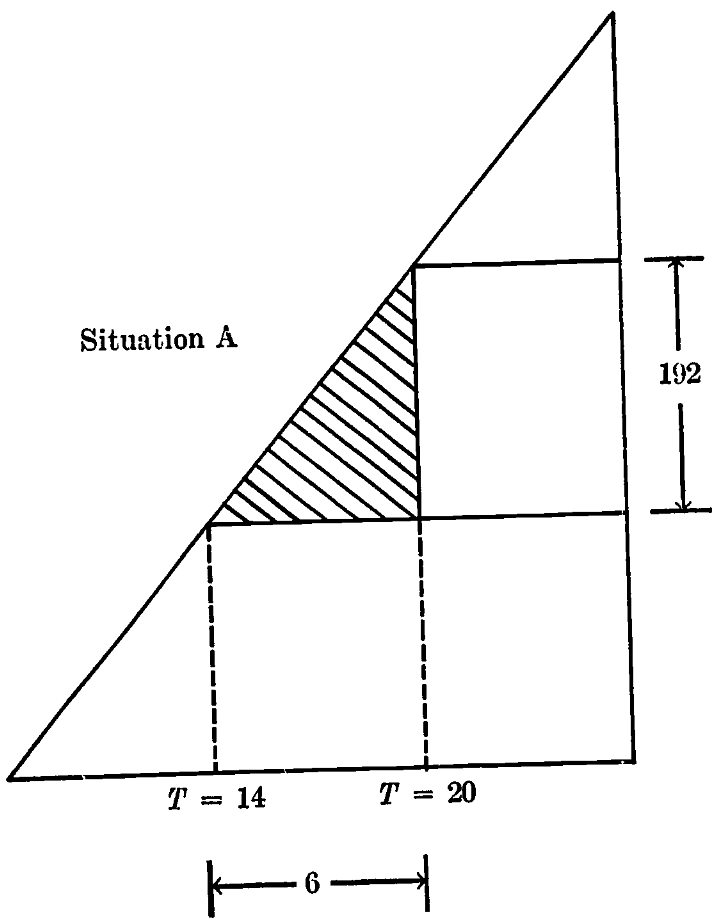
$$\text{Velocity} = 1\left(\frac{0}{2} + 32 + 32 + 32 + 32 + \frac{32}{2}\right) = 144$$

Integration 6. Time = 0 to Time = 6:

$$\text{Velocity} = 1\left(\frac{0}{2} + 32 + 32 + 32 + 32 + 32 + \frac{32}{2}\right) = 176.$$

The so-called intervals shown above are overlapping; that is, each one goes back to the beginning. This is to be expected, since the velocity at any time (say, Time = 5) is dependent on the preceding velocity values; that is, for any time, all previous (instantaneous) values make a contribution toward the next value, and so forth.

The first value above, 16, is only *one* value of the velocity during the interval from Time = 0 to Time = 1. (It can be shown that in this interval the velocity changed from 0 to 32.) The second value above, 48, again represents *one* value



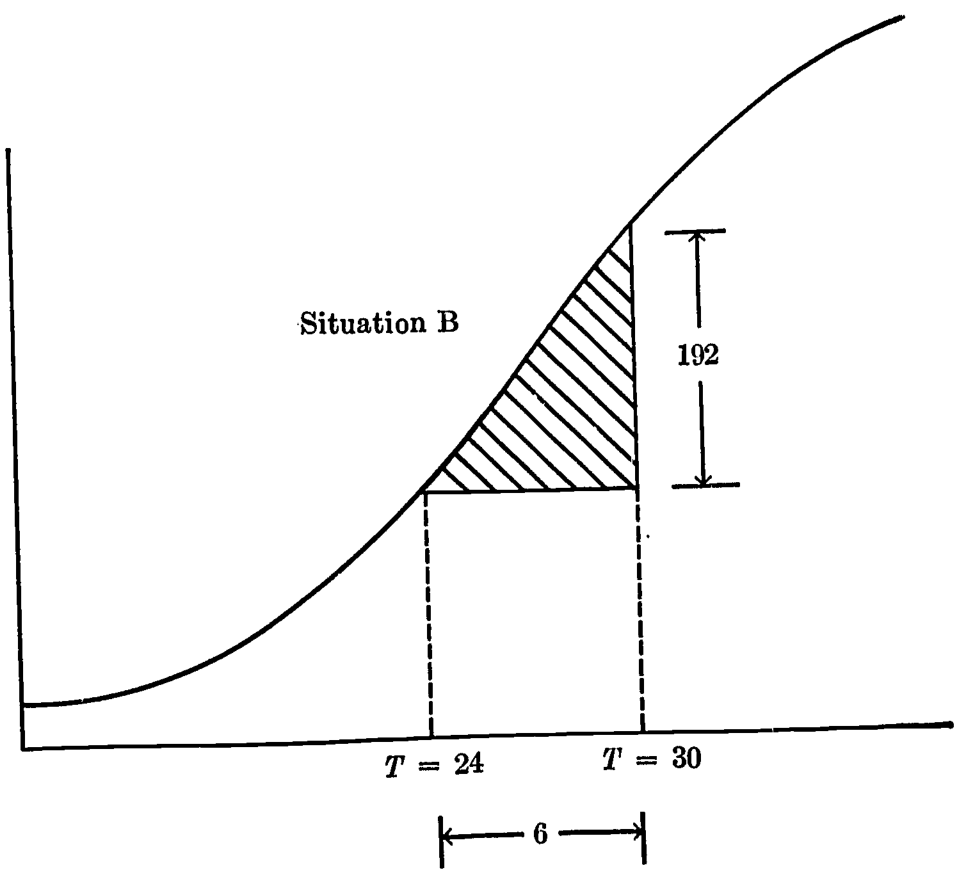


Fig. 3-18. Two Kinds of Time-Acceleration-Velocity Relationships

during the interval from Time = 1 to Time = 2 and takes into account the fact that at the start of this second interval the velocity was already 32. The computed values are then entered as midvalues in the tabulation below.

Time	Distance	Velocity*	Acceleration
0	0	0	0
		16	
1		(32)	32
		48	
2		(64)	32
		80	
3		(96)	32
		112	
4		(128)	32
		144	
5		(160)	32
		176	
6			32

* Values in parentheses are explained below.

Given the above mid-values, it is convenient for demonstration purposes to determine the values of velocity at the end of the intervals. For example, since 16 is the mid-value for the first interval and 48 is the mid-value for the second interval, it is reasonable to assign the average of 16 and 48, or $(\frac{1}{2})(16 + 48) = 32$, to the end of the first interval. All values in parentheses are obtained in this manner.

It is now possible to use the trapezoidal rule to obtain the values of distance by letting $h = 1$ and integrating the following values for velocity, just obtained:

Time	Velocity
0	0
1	(32)
2	(64)
3	(96)
4	(128)
5	(160)

Integration 1. Time = 0 to Time = 1:

$$\text{Distance} = 1\left(\frac{0}{2} + \frac{32}{2}\right) = 16$$

Integration 2. Time = 0 to Time = 2:

$$\text{Distance} = 1\left(\frac{0}{2} + 32 + \frac{64}{2}\right) = 64$$

Integration 3. Time = 0 to Time = 3:

$$\text{Distance} = 1\left(\frac{0}{2} + 32 + 64 + \frac{96}{2}\right) = 144$$

Integration 4. Time = 0 to Time = 4:

$$\text{Distance} = 1\left(\frac{0}{2} + 32 + 64 + 96 + \frac{128}{2}\right) = 256$$

Integration 5. Time $=$ 0 to Time $=$ 5:

$$\text{Distance} = 1\left(\frac{0}{2} + 32 + 64 + 96 + 128 + \frac{160}{2}\right) = 400\,.$$

By considering velocity mid-values, the original tabulation can be completed as shown below. (That is, an average velocity of 16 for the first second would give a distance of 16. With an average velocity of 48 for the next second, a distance of 48 would be added, or, $16 + 48 = 64$.)

Time	Distance	Velocity	Acceleration
0	0	0	0
		16	
1	16	(32)	32
		48	
2	64	(64)	32
		80	
3	144	(96)	32
		112	
4	256	(128)	32
		144	
5	400	(160)	32
		176	
6			32

Thus the value 16 for distance means that the velocity at Time $=$ 1 was 32 and, the average velocity was 16 over the interval—therefore the distance over the interval was also 16.

It is interesting to note that if the completed table, as developed by two successive integrations of acceleration, is differenced from left to right, the values for velocity are found to be equal to successive differences of the values of distance. In like manner, the values for acceleration are seen to be equal to the differences of successive values of the velocity.

Summary

This chapter contained illustrations of repetitive procedures. Some examples demonstrated techniques actually employed by persons who use digital computers. All examples suggested repetitive procedures and therefore were appropriate to the objective of the chapter: to demonstrate the use of repetitive procedures in problem solving.

The key idea in problem solving with automatic digital computers is repetition. If a problem has no repetitive characteristics, it is usually easier and faster to solve it by hand than to program it for a digital computer. Conversely, if enough repetition is involved in the solution of the problem, the use of modern high-speed computing devices is probably the only practical approach.

Two types of repetition are recognized in analyzing problems to be solved with digital computers. One type involves *repetition of data*. An example of data repetition is the calculation of interest on savings accounts in a bank. Here, the same relatively simple calculation is repeatedly performed on data that changes from account to account, from month to month.

The second type of repetition is usually called *iteration*. It is a method of approaching the solution of a problem by (1) making an estimate of the solution, (2) evaluating the estimate, (3) developing a new estimate on the basis of the evaluation, and (4) repeating the cycle again and again until a desired solution is reached. In some respects, methods based on iterative techniques are cut-and-try ones—they may seem to be lacking in the exactness associated with mathematics. Also, as has been indicated, under some conditions involving many calculations, it is not certain that the numbers used will produce the "correct" solution. All this points out that persons who use computers must (1) be aware of possible difficulties in using numerical methods and (2) understand the importance of numerical analysis in computer-oriented mathematics as it relates to topics such as convergence as well as the generation and analysis of errors.

References

For further detail on the subjects treated in this part of Chapter 3, the reader should consult the following references, which are listed in the Bibliography, Appendix C: 2, 14, 15, 16, 17, 20, 23, 32, 34, 35, 47, 52, 56, 59, 60, 64, 71.

EXERCISES

1. Figure 3-6 shows a procedure for extracting square root. The process continues until the test for the absolute value of the difference being less than 0.0001 is satisfied. Modify the procedure so that the test is made not according to the absolute value 0.0001 but according to 0.001 of the number the root of which is being sought.

2. Construct a flow diagram for looking up the value of the sine of any angle between 0 and 360 degrees in a table that has entries only from 0 to 45 degrees.

3. The algebraic signs of the sine and cosine vary according to the quadrant. Construct a flow diagram that will determine the proper quadrant for all combinations of values such as $\sin x = -0.64279$ and $\cos x = +0.76604$.

4. Construct a flow diagram for the linear interpolation process in a table such as the following:

y	x
155	19033
156	19312
157	19590
158	19866
159	20140
160	20412

In other words, determine what value of x corresponds to a given value of y, such as 158.26.

5. For the integers 1, 2, 3, 4, 5, 6, 7, 8, and 9, corresponding to x, compute values of y according to $y = x^2$. Now take the difference of the adjacent values thus obtained to produce the progression of odd numbers 1, 3, 5, 7, 9, etc. If the differences themselves were differenced, all of the values would be constant. Likewise, if the differences were found once more they would all be zero. Thus

it is that, for a polynomial of degree n, the $(n + 1)$th differences are zero. That is, for $x^2 (n = 2)$ the $n + 1$ or third differences are zero.

NOTE. If a column of differences is constant, the next higher column will contain all zeros and it will not be necessary to write them down. However, errors in data may introduce a "roughness" in the differences which can be used, by working backwards, to locate the error.

(a) Given a set of values of a polynomial, such as 166375, 175616, 185193, 195112, 205379, 216000, 226981, 238328, 250047, and 262144, determine the degree of the polynomial by taking differences. (b) Prepare a flow diagram to test for the degree of the polynomial where values are given.

6. Exercise 5 refers to the fact that the $(n + 1)$th differences of an nth degree polynomial are zero. Assuming that the following data are obtained from a source known to be represented by a polynomial of a given degree, construct a flow diagram to examine the data for the purpose of testing whether or not the data are correct. That is, perhaps there has been an error in data transmission or transcription. The data are: 29791, 32768, 35937, 39304, 42875, 46566, 50633, 54872, 59319, 64000, 68921, 74088.

HINT. Correct data will have "smooth" differences.

7. (a) Construct a flow diagram for the evaluation, by Simpsons' rule, of the expression $y = x^2 + 2$ between $x = 0$ and $x = 6$ (see graph in Fig. 3–17). (b) In Example 7, Part C, values of y are given for several values of x. At the end of Part C it is suggested that smaller differences between adjacent values of x can be selected to improve the accuracy of the calculation. Modify the diagram called for in (a) above to include an adjustment of the interval size (for example, divide it by 2). Include also a means of comparing the two values of the integral thus obtained. If the values are not different by more than 0.0005, stop. Otherwise, divide the interval once again and continue the process until the difference between consecutive trials is equal to, or less than, 0.0005.

Chapter 4

NONNUMERIC REPETITIVE PROCESSES

The digital computer is a powerful device for performing numerical calculations, since it is able to carry out arithmetic operations very rapidly. Some digital computer applications of this kind were discussed in Chapter 3.

The digital computer, however, is not limited to this kind of work, which is largely arithmetic. It is also a very powerful device for solving problems that may be thought to depend more on the manipulation and arrangement of data than on arithmetic processes involving the data.

These two areas of computer application are not completely separate since, on the one hand, a purely "mathematical" problem usually requires some arrangement or manipulation of data and, on the other hand, a purely "data processing" problem usually involves some arithmetic.

A way of distinguishing between the two general classes of digital computer applications is to note that mathematical problems may involve comparatively few numbers to start with but require extensive and complex operations to be performed on them. On the other hand, data processing problems may require comparatively simple operations but involve a very great many numbers.

Each class of problems has its own special requirements, difficulties, and techniques. It is not appropriate to attempt a comparison of them on the same scales of difficulty, efficiency, or speed, just as it is inappropriate to compare the difficulty of solving a 20 by 20 set of equations with the difficulty of preparing a payroll for 25,000 people. These are different classes of problems.

This chapter discusses some applications that depend less on arithmetic operations than on manipulative operations. The intention is to show that by the clever use of a few computer operations much useful work can be done.

A general approach that facilitates the solution of certain kinds of problems is based on a very simple idea: Compare two quantities and, depending on the result, proceed in one of two directions to the next step—where the process is repeated, starting with another comparison. This can be demonstrated by the following example.

Suppose it is desired to find out which person in a group is the tallest. Applying the above procedure, start by taking any two individuals and selecting the taller. The shorter of the two is eliminated, the taller one is compared with a new person, and the cycle is repeated. Eventually the process will eliminate all but the tallest person. Figure 4-1 illustrates the process, assuming that the two persons, A and B, are already selected. The diagram does not cover certain possibilities, such as what should be done if two persons have the same height

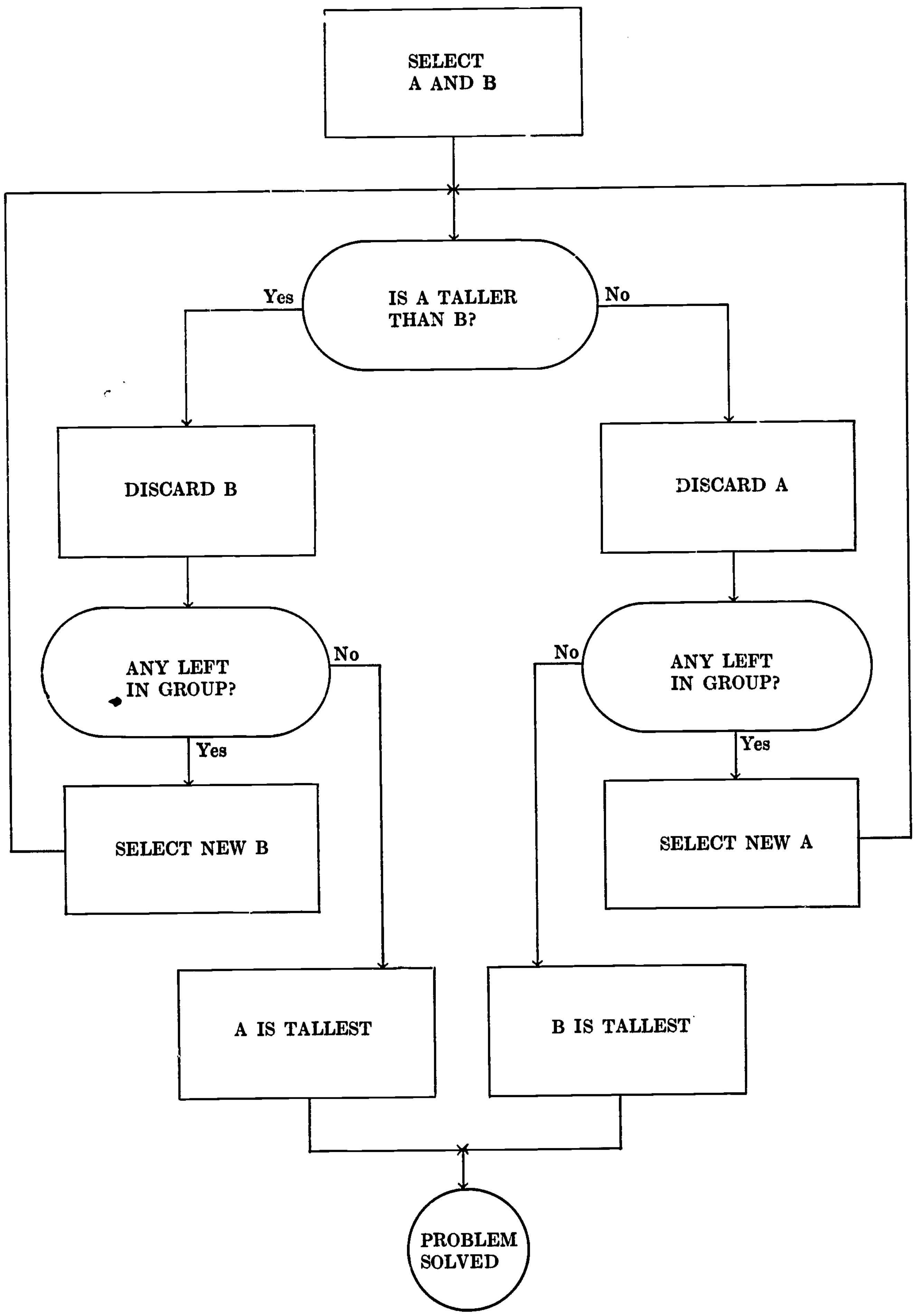


Fig. 4-1. Finding the Tallest Person in a Group

or when the last person has been selected. To take care of such possibilities the flow diagram in Fig. 4-1 would have to be modified.

The clever use of this simple idea provides an approach to many problem-solving situations. It is interesting to consider the possibility that all decisions—and, therefore, all solutions to all problems (for example, the buying of a house, the choosing of a wife, the planting of a garden, or the solving of a mathematical problem) can be reduced to a procedure similar to the one above.

Example 1. An Elimination Process

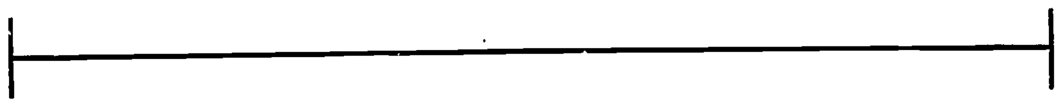
Consider some of the possibilities of problem solving by the less-than/greater-than/equal-to procedure.

One application of this procedure is in a game called "Twenty Questions." In this game one person thinks of an object and a second person attempts to find out what the first person is thinking of by asking not more than 20 questions which are answered either "yes" or "no." If the person asking the questions is clever enough to classify his questioning properly, he has a very good chance of winning every time simply because 20 yes-no decisions can cover a very large number of items.

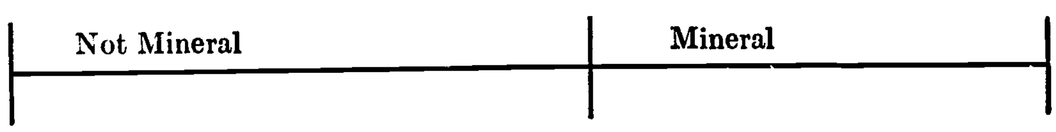
After the first question is answered all possible objects are classified into one of two classes. For example, if the question is "Is the object mineral?" and the answer given is "No," then one question eliminates all minerals. Notice that this is a much better question than "Is the object a pin?" because if the answer is "No," only all pins are eliminated as a possibility. Since *minerals* is a wider classification than *pins*, the questioning is not approaching an answer as rapidly as it could.

Notice that with one question it is possible to divide all things into two classes—the "Yes" and the "No." The second question will divide one of these classes into two others, etc.

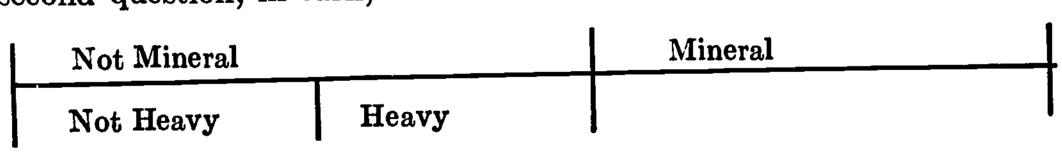
Another way to look at this is to consider the class of all objects as a line.



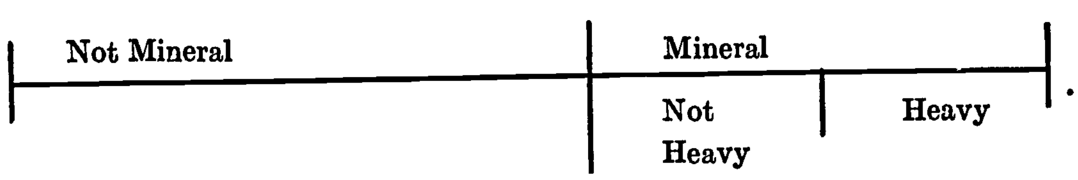
The first question essentially divides this line into two parts:



The second question, in turn, divides one of these sections again.



Or, if the circumstances had been different,



In other words, with two questions ("Mineral?" and "Heavy?") a total of four classifications is possible:

1. Mineral, heavy
2. Mineral, not heavy
3. Not mineral, heavy
4. Not mineral, not heavy.

A convenient way in which to express this is to let each question correspond to a position in a binary numeral. For example, let the first position correspond to "Mineral" and the second to "Heavy." If the answer is "Yes" represent it by a 1; if "No," by a 0.

"Heavy?" 2nd position	"Mineral?" 1st position	Means
0	0	Not heavy, not mineral
0	1	Not heavy, mineral
1	0	Heavy, not mineral
1	1	Heavy, mineral

Observe that this is the same sequence as the first four values in a binary notation system, as discussed in Appendix A. If a third category, such as "Expensive?" is considered, there will be eight classifications:

"Expensive?"	"Heavy?"	"Mineral?"
0	0	0
0	0	1
0	1	0
0	1	1
1	0	0
1	0	1
1	1	0
1	1	1

This is the same sequence as the first eight values in a binary notation system. In general, the number of classifications is equal to 2^Q, where Q stands for the total of what has been referred to here as "questions." During this process, of course, classifications are being excluded with each selection. The number of classes excluded is equal to $2^N - 1$ where N is equal to the number of the question being asked. For example, if there are three questions, there will be $2^3 = 8$ classifications. Also, on the third question $2^3 - 1 = 7$ classifications will have been eliminated, leaving one. In tabular form:

	Classifications (2^Q)	Exclusions ($2^N - 1$)
1	2	1
2	4	3
3	8	7
4	16	15
5	32	31
.		
.		
.		
20	1,048,576	1,048,575.

This shows that the procedure can be efficient in eliminating items that are not required. This is another way of saying that it is efficient in selecting items that are required. Suppose the above procedure is applied to a four-question game instead of a twenty-question game, and that the question is: "What number am I thinking of from 1 to 16, both inclusive?" (Suppose the number is 12.) According to the above table, this should be answered by four questions. For explanation, note the numbers given after the questions below.

Let question No. 1 be:
"Is the number greater than 8?"—Answer: "Yes"
Let question No. 2 be:
"Is the number greater than 12?"—Answer: "No"
Let question No. 3 be:
"Is the number greater than 10?"—Answer: "Yes"
Let question No. 4 be:
"Is the number greater than 11?"—Answer: "Yes"
Problem solved.

```
 1
 2
 3
 4
 5
 6
 7
 8 ———————— First midpoint
 —
 9
10 ———————— Third midpoint
 —
11 ———————— Fourth midpoint
 —
12 ———————— Second midpoint
 —
13
14
15
16
```

Notice that in this example, where the numbers were consecutive, each time the choice was made around the midpoint of the remaining group, only four questions were required to select from $2^4 = 16$ items.

Notice the similarity between this process and the earlier example to compare the heights of individuals. They are both methods of making a comparison between two items and on the basis of the result discarding something, and then making another selection as a basis for the next comparison, etc.

This comparatively simple process can be applied to many data processing problems since it provides a way for the digital computer to examine a list and select a particular value from the list. It is similar to the use of a numerical table, such as a sine or square root table. When an individual uses a sine table to determine the sine of 49 degrees, for example, he does not start at 0 degrees and proceed one-by-one until he locates 49 degrees. He will probably enter the table somewhere near the middle and, by a series of successive smaller "jumps," arrive at the desired value. The following example shows a computer application of this technique.

Example 2. Sorting by Seniority

In a certain company all employees are given a bonus at the end of the year. The amount of each bonus payment is determined by a separate calculation procedure corresponding to the number of years of employment. All employees who have worked for 16 years or more get the same bonus, and all are considered to have worked at least one year. The computer program calculates the number of years of employment and, as expressed in the COMPIAC compiler language, stores this number in a location called YEARS. The following program shows part of the bonus computation procedure. It would come after that part of the program in which the length of employment for an individual had been stored in the location called YEARS.

```
START BONUS CALCULATION:
  YEARS > 8: TWELVE. FOUR.
TWELVE:
  YEARS > 12: FOURTEEN. TEN.
FOURTEEN:
  YEARS > 14: FIFTEEN. THIRTEEN.
FIFTEEN:
  YEARS > 15: BONUS 16. BONUS 15.
TEN:
  YEARS > 10: ELEVEN. NINE.
ELEVEN:
  YEARS > 11: BONUS 12. BONUS 11.
THIRTEEN:
  YEARS > 13: BONUS 14. BONUS 13.
NINE:
  YEARS > 9: BONUS 10. BONUS 9.
```

The above steps would be followed by other parts of the program with the following labels, each one of which would carry out the proper calculations for a given length of employment. Notice that the steps for years less than eight have not been included.

```
BONUS 16:
BONUS 15:
BONUS 14:
etc.
```

Example 3. Sorting for Sequence

An important use of a comparison technique is that of sorting numbers into a particular order. This is a regular part of many data processing functions where it becomes necessary to arrange items such as payroll numbers or check numbers into an ascending sequence.

There are many ways of sorting numbers, and the following method serves as an example. Suppose the first few numbers stored in the computer are in the following order, and that any one number appears only once:

41
18
6
2
19
1
62
110
3
1006

The procedure follows:

Select the first two numbers and call them A and B:

$$A = 41$$
$$B = 18$$

Q. Is A greater than B? A. Yes.
(If A is greater than B, they are not in ascending order.)

Replace A with next number and try again:

$$A = 6$$
$$B = 18$$

Q. Is A greater than B? A. No.
(If A is not greater than B, they are in ascending order, but there may still be a number smaller than what is now called A.)

Keep this new A and replace the B with the next number:

$$A = 6$$
$$B = 2$$

Q. Is A greater than B? A. Yes.
(Again, not in order.)

Replace A with the next number:

$$A = 19$$
$$B = 2$$

Q. Is A greater than B? A. Yes.
(Same as before.)

Replace A:

$$A = 1$$
$$B = 2$$

Q. Is A greater than B? A. No.
(As before, maybe *this* is the smallest.)

Replace B:

$$A = 1$$
$$B = 62$$

Q. Is A greater than B? A. No.
Replace B:

$$A = 1$$
$$B = 110.$$

It is not necessary to continue to demonstrate what is happening. The process of replacing A whenever it is greater than B and by replacing B whenever it is greater than A results in the smallest number being "worked" into A and never being replaced.

The only additional thing which needed to be done is to ask, whenever any replacement is made for either A or B: "Is this the last number to be concerned with?" As long as the answer is "No," the process is continuous. Whenever it is "Yes," one more comparison needs to be made (with this last number). After this, whatever remains in A is the smallest number of the group, if this last answer is also "No." If this last number is "Yes," B has the smallest number of the group.

One must now simply store this smallest number in a special spot, and eliminate it from the list, thus:

New list	Storage
41	1
18	
6	
2	
19	
62	
110	
3	
1006	

Now begin again, as before, the only difference being that the smallest number of this cycle is stored in the next position of storage (not on top of the last one). This general procedure is an iterative one that can be used to arrange numbers into an ordered sequence.

Fig. 4-2 shows some of the most important steps in this process. There are certain steps left out of the diagram, some of which are referred to in the text, an important example being the process of storing the smallest number of each cycle.

Example 4. An Information Search

One method of filing reports and searching for them in a library consists of the following scheme, which makes use of comparison techniques. Before the system is installed, a list of words is selected which can be used to describe the expected reports. For example, in an aircraft plant the words might include "wing,"

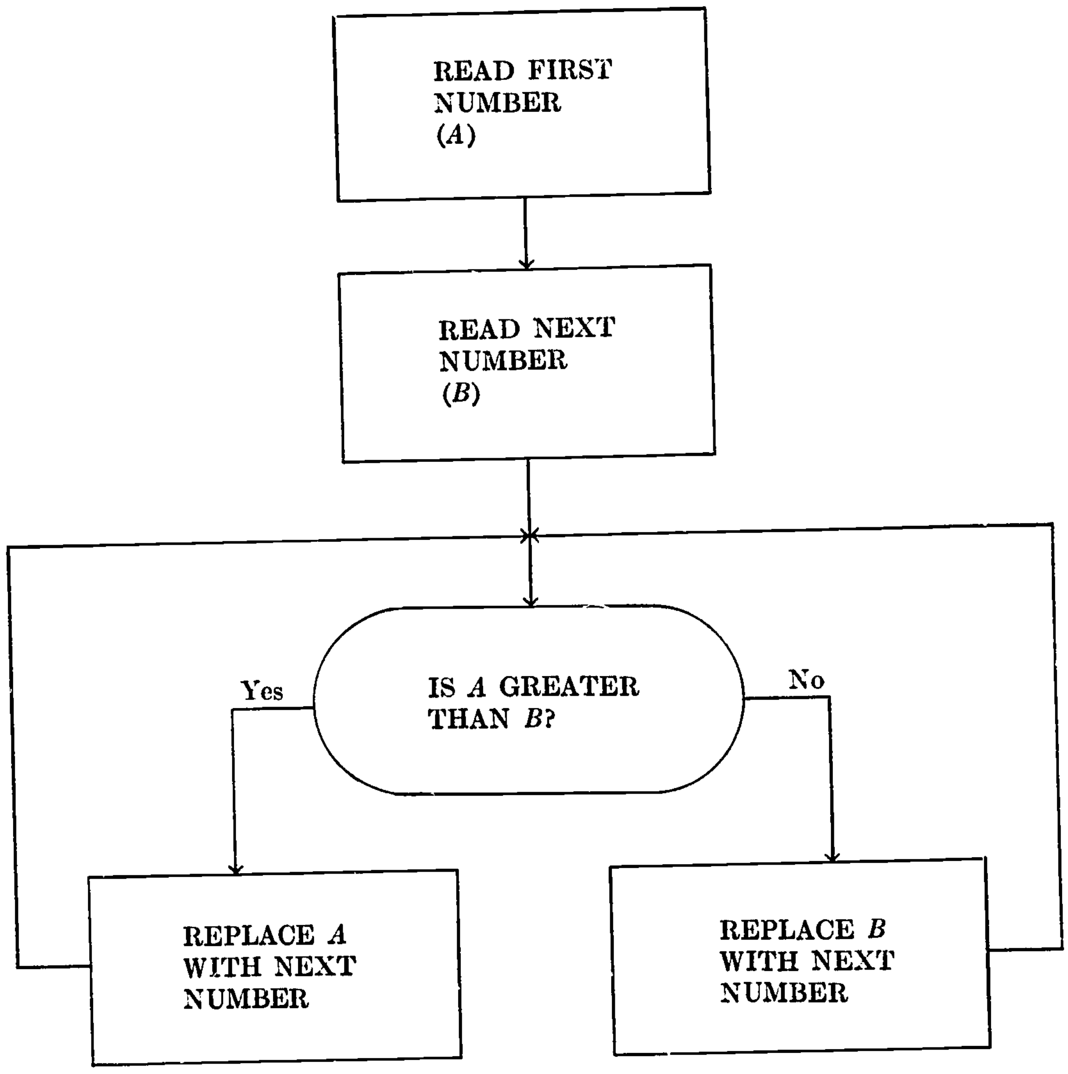


Fig. 4-2. Arranging Numbers in Ascending Sequence

"tail," "engine," "jet," "cockpit," "automatic," "pilot," "radio," "electronic," and of course many more. Each word would appear on an individual card of the type shown symbolically in Fig. 4-3. There may be several thousand words in the lists and new ones can, of course, be added as required. As reports are written, the filing process consists, first, of assigning each report a serial number. The report is then checked by the cataloger and that particular serial number is written down on the cards corresponding to as many of the words in the list as seem to apply. Suppose, for example, Report No. 1632 describes certain electronic devices by means of which a pilot can check on engine performance. Out of the words given above as samples, 1632 can be written on three cards: "Engine," "Pilot," and "Electronic." If Report No. 1633 concerned wing vibrations caused by jet engines, 1633 can be put on the following three cards out of the sample list above: "Jet," "Engine," and "Wing." Note that the card with the word "Engine" now has two numbers. In general, then, all reports which have anything to do with engines will have their numbers on the "Engine" card and so on.

To use the system to locate reports, or even general information if the name (or existence) of a report is unknown, cards are selected from the file which describe the area of interest, and a comparison is made. Suppose one is interested

WING 46 92 176 328 1633 1926	COCKPIT 99 142 887 1236 1901	PILOT 77 61 132 1632 1996
TAIL 48 162 897 1164	AUTOMATIC 44 72 86 990 1632 1996 2040	ENGINE 6 42 1531 1632 1633 1996
RADIO 19 28 1062 1909 2621	JET 9 206 384 1633 1746	ELECTRONIC 13 200 1632 1996 2006

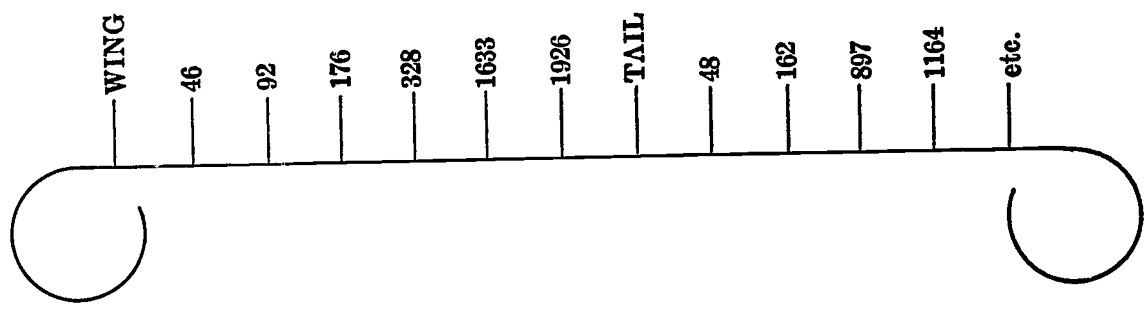
Fig. 4-3. Classification by Key Words

in automatic electronic devices by means of which the pilot can check engine performance. The following cards are pulled from the files: "Automatic," "Electronic," "Pilot," and "Engine." These are then examined visually and any serial number which appears on all cards *may* be that of a report pertaining to the subject specified. These reports are then pulled from the files and examined.

The above process of selection is one involving comparisons and depends on a definite repetitive procedure. The cards illustrated in Fig. 4-3 will clarify the above example, by showing what the cards may look like. Each number on a given card means that the particular report had something to do with the topic on the card. In other words, Reports 46, 92, 176, 328, 1633, and 1926 all had something to do with wings; Reports 77, 61, 132, 1632, and 1996 all had something to do with pilots; etc. For the particular example above, using the selected cards, notice that the circled numbers appear on more than one card.

Realize that a person using the system would not know what reports are referred to on any card; he would merely select the cards he thinks may contain references to reports of interest and subsequently make comparisons for numbers that appear on the selected cards.

This procedure has been applied to computing machines where, instead of the cards, the numbers as well as descriptive words are recorded on magnetic tape, as schematically shown below.



To use the system, a comparison method similar to that already described is used.

Without specifying the details, consider as before that the words ENGINE, ELECTRONIC, AUTOMATIC, and PILOT define the search to be made and that they are placed in the machine. These search words can be placed in the machine in any order, although there would be a time advantage if they were in the same order as the groups appear on the tape. The first one, ENGINE, is then placed in a position similar to what was called A in the example on sorting. At this point the tape is caused to move, and the entire first group is brought from the tape into the machine. (This group corresponds to the WING card.) The first information of the group (the word WING) is then placed in what was called B in the sorting example, and a comparison is made for the condition of equality, as illustrated in Fig. 4-4. In this figure there are steps omitted from the diagram similar to those left out of the example on sorting. It is intended to show how groups of data may be treated. Essentially the first word, ENGINE, is compared as follows:

Read ENGINE (A)
Read WING (B)
Q. Are they equal?
A. No.
(The WING group is not needed.)

Replace (B), but retain ENGINE (A) to compare with next group:

Holding ENGINE (A)
Read TAIL (B)
Q. Are they equal?
A. No.
(Same as before.)

Replace (B), but retain (A) once more, continuing the procedure until:

Holding ENGINE (A)
Read ENGINE (B)
Q. Are they equal?
A. Yes.
(This group is needed.)

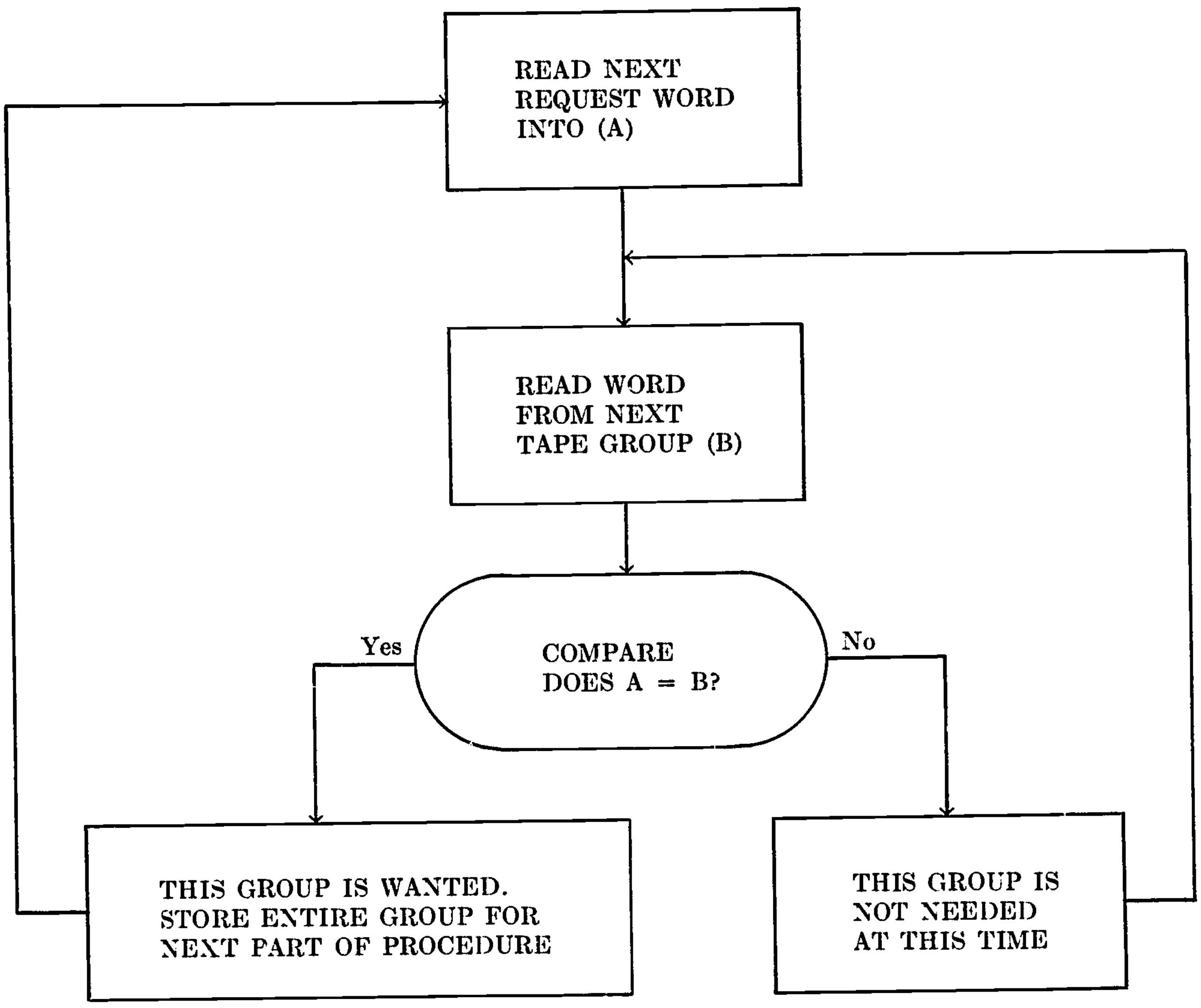


Fig. 4-4. Flow Diagrams for Group Selection

Take whole ENGINE group from tape and save it for the next part of the procedure.

Replace ENGINE (A) with ELECTRONIC. (This is the second search word.)

Start over with (B):
 Read ELECTRONIC (A)
 Read WING (B)
 Q. Are they equal?
 A. No.

Replace (B):
 Holding ELECTRONIC (A)
 Read TAIL (B)
 Q. Are they equal?
 A. No.

Replace (B).
 Etc.

The procedure will be continued where each of the search words will be compared with the group words on the tape and the appropriate groups removed from the tape for the next phase.

At this stage four groups have been removed from the tape: ENGINE, ELECTRONIC, AUTOMATIC, and PILOT. In a manner similar to that already described, two groups are selected for treatment, say the ENGINE group and the ELECTRONIC group.

This then puts the following sets of numbers into position.

ENGINE	ELECTRONIC
6	13
42	200
1531	1632
1632	1996
1633	2006
1996	

In a fashion similar to the sorting example, Item 6 from the ENGINE group, is compared successively as follows with the numbers of the ELECTRONIC group:

$$\frac{6}{13}\quad\frac{6}{200}\quad\frac{6}{1632}\ \text{etc.}$$

then 42 is compared

$$\frac{42}{13}\quad\frac{42}{200}\quad\frac{42}{1632}\ \text{etc.}$$

Finally the equal comparisons are located. They are 1632 and 1996. These two report numbers, then, are the only ones that need to be compared with any of the other remaining groups, and they are the only ones in the present example that will "survive" the comparison process as it is carried to completion. For example:

"Survivors from Last Comparision"	AUTOMATIC
1632	44
1996	72
	86
	990
	1632
	1996
	2040

As before, comparisons are made:

$$\frac{1632}{44}\quad\frac{1632}{72}\quad\frac{1632}{86}\quad\frac{1632}{990}\ \text{etc.}$$

and the same with 1996.

There will be two survivors out of the AUTOMATIC group, so the comparison is extended to the PILOT group, as follows:

"Survivors"	PILOT
1632	77
1996	61
	132
	1632
	1996

It can be seen, then, that Reports 1632 and 1996 are the only ones that apparently have something to do with subjects that can each be described by the words "automatic," "electronic," "pilot," and "engine." These reports can then be pulled from the file and examined. This library searching procedure is characteristic of many now under development or in use in various parts of the country.

Example 5. Making a Computer Compose Music

In Chapter 3 there was a discussion of the use of random numbers and some of their uses in solving problems. A computer application, which will be classed as nonnumerical, is that of music composition. This example will not cover all of the details of the process but will suggest a procedure.

When a musician composes a tune he somehow extracts information from his past experience and on his work sheet proceeds to write down a new note to follow the last note. This is a repetitive process although it is one that cannot be well defined and therefore not one that can be programmed for a computer. It is possible to simulate this process however.

One way to simulate the experience the composer has in his brain is to examine what he has already done. This can give some hint as to how he thinks. Suppose, then, we examine a number of tunes which have been composed by a given composer and which are more or less of the same recognizable type. Suppose also they are all in the same key. The examination might consist of tabulating the notes of all tunes in the group and counting how many times certain notes followed others. For example we might find that 76 percent of the time this composer followed the note C by the note G, that 22 percent of the time C was followed by E, and that 2 percent of the time C was followed by D. Similar tabulations would be prepared for all notes in a real application.

These tabulations represent one expression of the composer's experience. The problem to be solved is: How to make use of this experience in composing a new tune that will reflect this experience?

One way to do this is to have a list of numbers available the range of which is between 00 and 99, both inclusive, but which are in a random sequence. That is, their arrangement is of such a character that it corresponds to their having been selected by chance.

According to the tabulation, C will be followed by G, E, or D, with probabilities of 0.76, 0.22, and 0.02 respectively. This then can be represented by letting 00 and 01 correspond to a D following C, by letting 02 through 23 correspond to an E, and by letting 24 through 99 correspond to a G.

Now to compose a tune. Assuming that the note C has already been written, what will the next note be? The procedure is to select a number at random and see where it falls as defined by the "experience" probabilities for C. This having been done, repeat the process with a new random number according to the "experience" probabilities for G or E or D, whichever was selected.

The procedure above does not consider rests or rhythm, which may depend on "experience" developed by the examination of several preceding notes instead of the immediately preceding note. This brief procedure is intended only to give a suggestion of a music composition process. Some elements of a computer program to carry out the foregoing simplified procedure follow: GENERATE stands for the statements necessary to obtain a random number and store it in the location called RANDOM. PRINT stands for the necessary statements to print the letter specified.

```
RANDOM (1);
NOTE IS C:
  PRINT C,
  GENERATE,
  RANDOM [1] < 02: NOTE IS D. G OR E.

G OR E:
  RANDOM [1] < 24: NOTE IS E. NOTE IS G.

NOTE IS D:
  PRINT D,
  GENERATE,
  RANDOM [1] < ___: NOTE IS __. __ OR __.

NOTE IS E:
  PRINT E,
  GENERATE,
  RANDOM [1] < ___: NOTE IS __. __ OR __.

NOTE IS G:
  PRINT G,
  GENERATE,
  RANDOM [1] < ___: NOTE IS __. __ OR __.
```

The blanks in the last three parts of the program would, of course, be filled in according to the "experience" probabilities of those notes. In this example, each note is considered to be followed by one selected from a group in which there are not more than three notes. Also there would be additional parts of the program corresponding to all of the notes the composer had used. The entire procedure is simple in principle: After a note is selected, print it, and then on the basis of a chance process, but tempered by "experience," proceed to the next note and continue this procedure.

Example 6. Making a Computer Play a Tune

Example 5 suggests a related procedure: the generation of tones by computer. One way that this can be done by digital computer is to realize that between each two positions of the arithmetic unit there is a wire connection used to transfer the "carry" impulse. That is, between the third and fourth positions in a decimal machine, if the first three positions all contain 9's, the fourth will receive a "carry" when one more is added.

By making use of this carry facility it is possible to generate tones by computer, as follows:

Suppose that the wire that is used to transmit the carry impulse is instead connected through some external electronic circuits including a loud speaker system. Now if the carry impulse is transmitted, the speaker will "click" in a way similar to the way the television speaker may click when a house light switch is turned on.

Our ears recognize tones by their frequencies, therefore, if our ears detect "clicks" at, say, 440 times per second we will interpret this to be the tone A.

Is there a way, then, of using the computer to produce "clicks" at frequencies that correspond to tones of the scale, making use of the carry circuit?

Suppose there is a computer that can add numbers at the rate of 10,000 per second (which is actually a modest rate for modern computers). Suppose also that the carry circuit connecting the sixth and seventh positions of the counter is connected to the speaker. The speaker will now click every time the counter carries into the seventh position, or on every million. Therefore, by causing this computer to add repeatedly the number 44,000 for one second, this particular carry circuit would be engaged 440 times, since $44{,}000 \times 10{,}000 = 440{,}000{,}000$, and the loud speaker connected to the circuit would "click" 440 times per second, producing the tone A.

Before proceeding further it should be explained that tones of different lengths, quarter notes, half notes, etc., can be produced in the same way. Recall that the "clicking" does not have to continue for a second to produce A. It is the *rate* of clicking that distinguishes the tone. Thus, if 44,000 were added for one second it would produce 440 clicks and therefore A, but A would also be produced if this rate of clicking lasted only one-half second—it would merely be a tone of shorter duration. Therefore the duration of notes depends on when the process is stopped. This "stopping" process is illustrated for three notes in the program that follows.

To have the program "play a tune" the notes of the tune can be generated as suggested in Example 5, or entered directly by cards. Let the following notes, expressed in do-re-mi relationships, have these numerical codes:

Do whole note = 1
Do half note = 2
Do quarter note = 3
Re whole note = 4
Re half note = 5
Re quarter note = 6
Ti dotted half note = 7
Ti half note = 8
Ti quarter note = 9

Using this code, and letting A be do: the first 6 notes of *America*, starting at A, will be as follows:

2, 2, 5, 7, 3, 5.

For this example assume that these notes will be entered, in order, by cards into the computer in locations called NOTES. The elements of the program follow.

```
NOTES (6),
COUNTER (1);
I = 1 (1) 6
  {NOTES [I] < 4: NOTE IS DO. NOTE IS RE OR TI.

NOTE IS DO:
  NOTES [I] < 3: WHOLE OR HALF DO. START QUARTER DO.

WHOLE OR HALF DO:
  NOTES [I] < 2: START WHOLE DO. START HALF DO.

NOTE IS RE OR TI:
  NOTES [I] < 7: NOTE IS RE. NOTE IS TI.

NOTE IS RE:
  NOTES [I] < 6: WHOLE OR HALF RE. START QUARTER RE.

WHOLE OR HALF RE:
  NOTES [I] < 5: START WHOLE RE. START HALF RE.

NOTE IS TI:
  NOTES [I] > 8: START QUARTER TI. DOT HALF OR HALF TI.

DOT HALF OR HALF TI:
  NOTES [I] = 7: START DOT HALF TI. START HALF TI.

START WHOLE DO:
  0 → COUNTER [1],

MAKE WHOLE DO:
  COUNTER [1] + 44000 → COUNTER [1],
  COUNTER [1] = 440000000: RETURN. MAKE WHOLE DO.

START HALF DO:
  0 → COUNTER [1],

MAKE HALF DO:
  COUNTER [1] + 44000 → COUNTER [1],
  COUNTER [1] = 220000000: RETURN. MAKE HALF DO.

START QUARTER DO:
    0 → COUNTER [1],

MAKE QUARTER DO:
  COUNTER [1] + 44000 → COUNTER [1],
  COUNTER [1] = 110000000: RETURN. MAKE QUARTER DO.

RETURN:},
  STOP.
  STOP:..
```

To save space, only three of the nine notes are included here. The others would be developed in the same way using different values to correspond to the different frequencies. It should be noted that frequencies produced by this procedure will not be correct because the program is doing other things besides "adding to a million," such as selecting the proper note and checking to determine if it is time to stop the process. Therefore the number of carry impulses to the loud speaker will be the wrong number per second. This could be adjusted in actual practice, of course, by selecting other numbers instead of the 44,000 used here for A. For example if the "other things" being done by the computer require an amount of time equal to that needed to perform the addition, it would be necessary to add 88,000 instead of 44,000 to produce the required 440 carry impulses per second.

EXERCISES FOR CHAPTER 4

1. Example 2 showed a process that started in the middle of an ascending sequence of numbers and would select a particular one out of all cases having a value greater than this midpoint value. Prepare a flow diagram and write a program to continue the process for values that are less than the midpoint value. Do the same thing using "equal" instead of "less than" or "greater than." How does this change the procedure?

2. Example 3 showed a sorting method that resulted in an ascending sequence having the smallest number "on top." How would the flow diagram be changed to make the sequence a descending one? Write a compiler program to do this.

3. In Fig. 4-3 of Example 4 the report numbers within a given group had previously been arranged into ascending order. What is the advantage of this having been done before the searching process is begun?

4. In a four-year high school having 1200 students the grade point averages for all students are arranged in alphabetical order of the students' names, without regard to the class year of the students. Each grade point average is coded, however, to show the student's particular year in school. The code consists of a "1" for the first-year students, a "2" for the second year students, etc. A typical student record is of the form:

3205,

meaning that this pertains to a third-year student with a 2.05 average. Prepare a flow diagram and program that will arrange the now single alphabetically arranged list into four lists, one for each class year, in ascending order of grade point average.

5. How would the program of Exercise 4 be changed to count the number of students in each class?

6. Select 5 nursery rhymes and, after being sure that they are in the same key, prepare the experience probabilities as suggested in Example 5. It will be desirable to consider a "rest" the same as a note. Select random numbers by making a set of ten cards numbered from 0 to 9. Put these in a box and shake them. Select a card and record it as the units digit of a two-place number. Return the card to the box and shake again. Select a card again and record it as the tens position of the random number. Compose a tune and play it on the piano.

Appendix A

NUMBER NOTATION SYSTEMS

This appendix discusses three computer oriented topics. Although it is not necessary to master all the fine points of these topics in order to understand and use digital computers, it is desirable to broaden one's comprehension of the digital computer and its applications by supplementing the familiar decimal system of notation with certain others.

The three topics included in this section are: *octal, duodecimal,* and *binary notation,* used extensively in discussing the internal operations of many digital computers; *binary coded decimal,* a system used for representing symbols in binary code; and *floating point,* a system similar to scientific notation, which is used in most engineering and scientific computations.

Part I. Octal, Duodecimal, and Binary Notation Systems

In the Introduction the point was made that certain devices (computers) have been developed to aid in solving both mathematical and nonmathematical (or, more appropriately, numeric and nonnumeric) problems that have been properly organized in some sequential fashion.

One topic of interest and importance from both a mathematician's and digital computer engineer's point of view is the study of number notation systems. In the study of number theory one learns some of the basic properties of numbers. Among these properties is that the scale of notation may be arbitrarily selected; that is, our decimal system could be replaced by a different system of notation. Although it would be confusing to persons trained in the decimal system, another system would be equally adequate for solving problems.

Since a system of notation is essentially a means of counting, the term *different system* means counting by digits that are not repeated, or cycled, at 9, as is done in the familiar decimal system. That is, when we use 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, we start repeating at 10 in the sense that we use the same digits again. This is illustrated by the following table:

E	D	C	B	A
		0	0	0
		0	0	1
		0	0	2
		0	0	3
		0	0	4
		0	0	5
		0	0	6
		0	0	7

E	D	C	B	A
		0	0	8
		0	0	9
		0	1	0
		0	1	1
		0	1	2
		0	1	3
		0	1	4
		0	1	5
		0	1	6
		0	1	7
		0	1	8
		0	1	9
		0	2	0
		0	2	1
		0	2	2
		0	2	3
		0	2	4
		0	2	5
		0	2	6
		0	2	7
		0	2	8
		0	2	9
		0	3	0
		0	3	1
		0	3	2
		0	3	3
		0	3	4
		0	3	5
		0	3	6
		0	3	7
		0	3	8
		0	3	9
		0	4	0

Notice that in column A each of the digits is used once until all have been used once; then the cycle is repeated. In column B each of the digits is used ten times before the next one is used for ten times. Likewise, in column C each digit will be used 100 times before the next digit is used for its 100 times. In column D each digit would be used 1000 times; in column E, 10,000, and so forth.

The point to emphasize is that the numeral 10 can have another meaning than that of designating the total number of fingers, including thumbs, on both hands. The other meaning is that we can say "10" means "the place where the system starts repeating."

With this new meaning of "10" in mind, notice that in column A the digits change their value every $10^0 = 1$ times. That is, every single time there is a change in digits. In column B the change is every $10^1 = 10$ times. In column C the change would be every $10^2 = 100$ times, and so forth. This suggests that any number can be named using the decimal system of notation. That is, any number can be thought of as a sum of powers of the base 10 multiplied by an appropriate number from 0 to 9, inclusive.

For example, 1 2 3 7 can be represented as follows: the 7 corresponding to position A, the 3 to position B, the 2 to position C, and the 1 to position D, as follows:

D	C	B	A
1	2	3	7

or

$$1(10)^3 + 2(10)^2 + 3(10)^1 + 7(10)^0,$$

or

$$1000 + 200 + 30 + 7.$$

A more general representation of this is to say that any integer N can be written as follows:

$$N = C_n(10)^n + \cdots + C_2(10)^2 + C_1(10)^1 + C_0(10)^0$$

where the values are rewritten from left to right in order of decreasing powers of 10, and the C's represent any one of the digits, including zero.

Consider the second meaning of "10," as stated in the paragraph above: "the place where the system starts repeating." Assume that "10" means "the place where the system starts repeating" according to the total number of fingers on both hands, not including thumbs. With this system we would use: 0, 1, 2, 3, 4, 5, 6, 7. In other words, there are now as many symbols (called digits before) as there are fingers, not including thumbs.

As before, an array of numerals can be developed as follows:

C	B	A	Decimal
0	0	0	0
0	0	1	1
0	0	2	2
0	0	3	3
0	0	4	4
0	0	5	5
0	0	6	6
0	0	7	7
0	1	0	8
0	1	1	9
0	1	2	10
0	1	3	11
0	1	4	12
0	1	5	13
0	1	6	14
0	1	7	15
0	2	0	16
0	2	1	17
0	2	2	18
0	2	3	19
0	2	4	20
0	2	5	21
0	2	6	22
0	2	7	23
0	3	0	24
0	3	1	25
0	3	2	26
0	3	3	27
0	3	4	28
0	3	5	29
0	3	6	30
0	3	7	31
0	4	0	32

Notice that there is no difference between the general structure of the new system and the old system: they both cycle or repeat at the *numeral* "10." As can be seen, column A repeats each 8^0 times; column B repeats each 8^1 times; and column C, if extended, would repeat each 8^2 times. Thus in this system, known as *octal*, any integer can be represented as follows:

$$N = C_n(8)^n + \cdots + C_2(8)^2 + C_1(8)^1 + C_0(8)^0$$

where the C's again represent any one of the symbols 0, 1, 2, 3, 4, 5, 6, 7. Of course, the symbols 8 and 9 are no longer used in the new system. For convenience in the chart above, the corresponding decimal values are given for the octal numerals; for example, 32 in decimal notation equals 40 in octal notation, etc. It is customary to write this as follows:

$$(32)_{10} = (40)_8 .$$

The subscripts 10 and 8 designate the *base*, or *radix*, of this system. Notice that the base corresponds to the number of different symbols, expressed here in base 10, required in the system.

The duodecimal system of notation is shown below to indicate how a system can be developed on a base larger than "the number of fingers on two hands, including thumbs."

Duodecimal				Octal	Decimal
D	C	B	A		
		0	0	0	0
		0	1	1	1
		0	2	2	2
		0	3	3	3
		0	4	4	4
		0	5	5	5
		0	6	6	6
		0	7	7	7
		0	8	10	8
		0	9	11	9
		0	L (lam)	12	10
		0	Z (zug)	13	11
		1	0	14	12
		1	1	15	13
		1	2	16	14
		1	3	17	15
		1	4	20	16
		1	5	21	17
		1	6	22	18
		1	7	23	19
		1	8	24	20
		1	9	25	21
		1	L	26	22
		1	Z	27	23
		2	0	30	24
		2	1	31	25
		2	2	32	26
		2	3	33	27
		2	4	34	28
		2	5	35	29
		2	6	36	30

(*continued next page*)

Duodecimal D C B A	Octal	Decimal
2 7	37	31
2 8	40	32
2 9	41	33
2 L	42	34
2 Z	43	35
3 0	44	36

As before, any integer can be represented as follows:

$$N = C_n(12)^n + \cdots + C_2(12)^2 + C_1(12)^1 + C_0(12)^0.$$

There is something new, however. Instead of leaving out two symbols, 8 and 9, as in the octal system, two additional symbols must be added. These have been called here L (for lam) and Z (for zug). Lam and zug have no special meaning: they are merely arbitrary names for two arbitrary symbols. Their use also introduces new terms, such as *lamteen* and *twenty-zug*. Notice that the symbol "10" still means "the place where the system starts repeating." For comparison purposes, the octal and decimal systems are listed alongside the duodecimal. In particular, for example:

$$(2L)_{12} = (42)_8 = (34)_{10}.$$

From the computer engineer's point of view, there are certain advantages to still another notation system, which is based on two (the total number of thumbs on both hands).

A reason for wanting this system with a base equal to two is related to the fact that electronic devices can be developed which do a good job in distinguishing between the following kinds of situations, which are typical of conditions that can be experienced using electronic circuitry:

1. Is a switch *open* or *closed*?

or

2. Is a voltage *present* or *absent*?

In other words, these are situations in which there are only two conditions possible, not eight, not ten, and not twelve.

Therefore, to match the electronic devices which can only distinguish between two states or conditions, a binary (base = 2) notation system has certain advantages.

In principle, the binary notation system is no different from the others: "10" still means "the place where the system starts repeating," and any integer can be represented as follows:

$$N = C_n(2)^n + \cdots + C_2(2)^2 + C_1(2)^1 + C_0(2)^0.$$

As before, the following list will indicate how this system is used.

Comparison is given with the octal and decimal systems; for example:

$$(10000)_2 = (20)_8 = (16)_{10}$$

Binary					Octal	Decimal
E	D	C	B	A		
				0	0	0
				1	1	1
			1	0	2	2
			1	1	3	3
		1	0	0	4	4
		1	0	1	5	5
		1	1	0	6	6
		1	1	1	7	7
	1	0	0	0	10	8
	1	0	0	1	11	9
	1	0	1	0	12	10
	1	0	1	1	13	11
	1	1	0	0	14	12
	1	1	0	1	15	13
	1	1	1	0	16	14
	1	1	1	1	17	15
1	0	0	0	0	20	16

Notice that in column A the cycling is every 2^0 times; in column B, every 2^1 times; in column C, every 2^2 times; etc.

The purpose of the above material is to give some insight, or feeling, as to how systems of notation are "put together;" that is, what are their basic elements. Two related topics will be discussed below: (1) the extension of the above material to fractions and (2) the method of conversion from one system to another.

To extend the idea to fractions, first consider the integer:

$$(137)_{10}.$$

This can be represented as:

$$1 \times 10^2 + 3 \times 10^1 + 7 \times 10^0.$$

Likewise, a fraction can be similarly represented; for example:

$$137.642 = 1 \times 10^2 + 3 \times 10^1 + 7 \times 10^0 + 6 \times 10^{-1} + 4 \times 10^{-2} + 2 \times 10^{-3}.$$

In other words, the powers of 10 range from zero in a positive direction for integers and from -1 in a negative direction for fractions. The same situation prevails regardless of the base or radix in question.

The problem of conversion from one system to another can be thought of as a determination of the number of groups of 10 within the numeral in question. In this sense, of course, 10 has the meaning of "the number of items in a repetitive cycle for the given base" (not necessarily the number of fingers, including thumbs, on both hands).

For example, convert the numeral $(12)_{10}$ to its octal equivalent. In line with the paragraph above it is necessary to determine how many $(10)_8$'s are in $(12)_{10}$.

Of course $(10)_8 = (8)_{10}$ (see the conversion table, pages 138–39). Also, it can be stated that

$$(12)_{10} = (8)_{10} + (4)_{10}\,.$$

In other words, $(12)_{10}$ equals a single value of $(8)_{10}$ plus a remainder of $(4)_{10}$. But, since each $(8)_{10} = (10)_8$, the above expression can be written as

$$(14)_8\,.$$

In general, then, the system is the same as earlier described: The number is expressed as a sum of powers of other numbers plus a remainder.

In converting from decimal to octal, subtract out powers of 8 as follows:

$$\frac{\begin{matrix} C & & B & & A \end{matrix}}{8^2 + 8^1 + 8^0}\,.$$

For an 8^2 a 1 is placed in the C position; for an 8^1, a 1 is placed in the B position; and the remainder is placed in the A position. In the above example, $(12)_{10}$ contains an 8^1 plus a remainder of 4, hence

$$(12)_{10} = (14)_8\,.$$

In like manner,

$$(27)_{10} = (33)_8$$

since

$$(27)_{10} = 3(8)_{10} + 1(3)_{10}\,.$$

Also,

$$(64)_{10} = (100)_8$$

since $(64)_{10}$ is 8^2 and is recorded in the C column. Note then that:

$$(79)_{10} = 1(8)^2_{10} + 1(8)^1_{10} + 7(8)^0_{10}$$

or

$$(79)_{10} = (117)_8$$

and, again,

$$(142)_{10} = 2(8)^2_{10} + 1(8)^1_{10} + 6(8)^0_{10}$$

or

$$(142)_{10} = (216)_8\,.$$

Another way of approaching this problem is to perform successive divisions by the base to which one wishes to convert, and record the remainders. For example:

$$\begin{array}{rl} 8\underline{|142} & \text{Remainders} \\ 8\underline{|17} & +6 \\ 8\underline{|2} & +1 \quad \therefore (142)_{10} = (216)_8. \\ 0 & +2 \end{array}$$

This is essentially the same as the first procedure outlined since it is effectively removing powers of 8. After the first division by 8, the number can be considered as seventeen 8's plus 6. After the second division the number can be considered as two 8^2's, plus one 8^1, plus 6; thus the octal numeral 216 is obtained.

In the same way powers of 2 can be subtracted to convert from decimal to binary.

EXAMPLE. Convert $(18)_{10}$ to binary. Recalling that the powers of 2 are

$$\begin{array}{cccccc} 2^5 & 2^4 & 2^3 & 2^2 & 2^1 & 2^0 \\ 32 & 16 & 8 & 4 & 2 & 1, \end{array}$$

begin by subtracting out the largest power of 2, and record a 1 in the corresponding position. The largest power of 2 that can be subtracted from 18 is 2^4, or 16. This will call for a 1 to be placed in the fifth position and will leave a remainder of 2. From this remainder the highest power of 2 which can be subtracted is 2^1; it will call for a 1 to be placed in the second position, and it will leave no remainder. Therefore,

$$(18)_{10} = (10010)_2 .$$

As above, by successive division,

$$\begin{array}{r|l l} 2 & \underline{18} & \text{Remainders} \\ 2 & \underline{9} & +0 \\ 2 & \underline{4} & +1 \\ 2 & \underline{2} & +0 \quad \therefore (18)_{10} = (10010)_2 . \\ 2 & \underline{1} & +0 \\ & 0 & +1 \end{array}$$

The study of systems of notation is interesting and can develop an understanding of certain properties of numbers. From the point of view of the individual whose primary task is to solve mathematical problems with the aid of large digital computers, the subject is of little practical use. The reason for this is that few, if any, computers require familiarity with any but the decimal system, since even those which are fundamentally binary are usually used in conjunction with compilers and other programming procedures which perform the necessary conversions from one system to another.

These same techniques of conversion can be applied to the duodecimal system as well. The interesting aspect of conversion with this system and perhaps with others, too, is that a new multiplication table is required to perform the successive divisions. A table for the duodecimal system is given below. Remember that to convert from one base to another it is necessary to perform successive divisions by the base to which it is desired to convert. For example, convert $(100)_{12}$ to its decimal equivalent. Since lam (L) is the duodecimal equivalent of the decimal 10, division must be by L:

$$\begin{array}{r@{\,}r} & 12 \\ \mathrm{L} | & \overline{100} \\ & \underline{\mathrm{L}\ \ } \\ & 20 \\ & \underline{18} \\ & 4 \end{array} \qquad \begin{array}{r@{\,}r} & 1 \\ \mathrm{L} | & \overline{12} \\ & \underline{\mathrm{L}} \\ & 4 \end{array} \qquad \begin{array}{r@{\,}r} & 0 \\ \mathrm{L} | & \overline{1} \\ & \underline{0} \\ & 1 \end{array}$$

or $(100)_{12}$ equals $(144)_{10}$.

DUODECIMAL MULTIPLICATION TABLE

	1	2	3	4	5	6	7	8	9	L	Z	10
1	1	2	3	4	5	6	7	8	9	L	Z	10
2	2	4	6	8	L	10	12	14	16	18	1L	20
3	3	6	9	10	13	16	19	20	23	26	29	30
4	4	8	10	14	18	20	24	28	30	34	38	40
5	5	L	13	18	21	26	2Z	34	39	42	47	50
6	6	10	16	20	26	30	36	40	46	50	56	60
7	7	12	19	24	2Z	36	41	48	53	5L	65	70
8	8	14	20	28	34	40	48	54	60	68	74	80
9	9	16	23	30	39	46	53	60	69	76	83	90
L	L	18	26	34	42	50	5L	68	76	84	92	L0
Z	Z	1L	29	38	47	56	65	74	83	92	L1	Z0
10	10	20	30	40	50	60	70	80	90	L0	Z0	100

EXERCISES

1. Extend the table showing the duodecimal system to include bases 3, 6, and 14. (Make up your own symbols as needed.)

2. Convert the following from decimal to binary notation: (a) 36; (b) 151; (c) 378; (d) 2021.

3. Convert the following from binary to decimal notation: (a) 101; (b) 1100; (c) 111101; (d) 10010011.

4. Convert 77 in decimal notation: (a) to octal notation, (b) to binary notation.

5. Compare the octal and binary notation in the previous table and discover the meaning of the statement: "Octal is a shorthand for binary."

HINT. $2^3 = (8)_{10} = (10)_8$. In the top line of Exercise 6, below, separate the binary quantity into groups of three, starting at the right, to give 10 $\wedge$ 010. Converting each group of three gives 2 $\wedge$ 2.

6. In the following table, numerals having the same value are given in the same row with column headings that designate different systems of numeration. For example, decimal 18 is equivalent to octal 22, to binary 10010, or to duodecimal 16. Fill in the remaining blank positions.

Decimal	Octal	Binary	Duodecimal
18	22	10010	16
	37		
246			
	75		
	41		
			144
		1000	
100			
			3L
		10010000	

Part II. Binary Coded Decimal

One of the most frequent conversions performed by present-day computers is the conversion from decimal to binary notation. Another one is the conversion from decimal to what is called binary coded decimal—abbreviated BCD—which is described below.

Because such conversions must be performed often in the use of computers, programs are available for them. Of course, once the programming is done, it is not necessary to worry about the task from day to day. It is for this reason that a computer user does not need to have great facility in performing the conversions—the computer can be programmed to do this. However, the techniques are interesting to many individuals who teach or otherwise use mathematics. This material is presented for this reason, and also because an added appreciation for numerical relationships can be developed by acquiring some understanding of some of the systems of notation.

BCD (binary coded decimal) is a hybrid code of binary and decimal; that is, each separate decimal digit is represented in binary form. The paragraphs below describe the conversion of decimal to BCD.

In Part I of this appendix, the binary representations of the decimal digits 0–9 were shown to be as follows.

Decimal	Binary
0	0
1	1
2	10
3	11
4	100
5	101
6	110
7	111
8	1000
9	1001
10	1010

BCD uses one of the above binary representations for each decimal digit of a given numeral. There are certain advantages in handling some types of data if each decimal digit is handled separately. For example, the decimal numeral 28, in binary, is as follows:

$$(28)_{10} = (11100)_2 .$$

Notice, however, that the decimal digits have lost their identity in the binary form. In BCD, the arrangement is as follows:

$$\begin{array}{cc} 2 & 8 \\ 0010 & 1000 \end{array}$$

or

$$(28)_{10} = (00101000)_{BCD} .$$

Here each decimal digit is represented by a four-place binary numeral. As in the above examples, a division process can produce the conversion. In this case the binary quantity is divided by the binary equivalent of the decimal 10, or 1010.

```
          10              0
  1010|11100      1010|10
       1010               0
       ----              --
        1000             10
        0000
        ----
        1000
```

Where, as before, the last remainder (10) after filling out the four positions becomes 0010, which together with the remainder of 1000 produces 28.

Again, it should be emphasized that this merely shows an iterative process which may be done by machine, but is not a mathematical application that a machine user himself must perform. It gives some insight as to what is going on inside the machine as it converts from one binary representation to another. In actual practice a different procedure would be used . . . one that depends on principles like those described in the first 2 examples given in Chapter 4.

The following conversion of $(4962)_{10}$ to BCD extends the above procedure. First, by methods shown previously:

$$(4962)_{10} = (1001101100010)_2 .$$

Second, the conversion:

```
        111110000              110001
1010|1001101100010      1010|111110000
      1010                   1010
      10010                   1011
       1010                   1010
       10001                    10
        1010                     0
         1111                   100
         1010                     0
          1010                  1000
          1010                     0
            00                  10000
             0                   1010
            00                    110
             0
            01
             0
            10
             0
            10

        100                     0
1010|110001             1010|100
      1010                     0
       100                   100
         0
      1001
         0
      1001
```

and, rearranging the remainders, one obtains 0100, 1001, 0110, 0010, or, in decimal notation, 4962.

A BCD code can also be used for other than numeric symbols. Reference to Fig. 2-3 shows that the letter A corresponds to a "1" in combination with a hole in the upper level of the card. (This upper level of the card is sometimes called "Y".) If the Y level is given the binary code 11; if the middle level, called "X", is given the binary code 10; and if the lower level, called "zero", is given the binary code 01; the *alphabetic* BCD can be shown to be as follows:

A	11 0001	J	10 0001	S	01 0010
B	11 0010	K	10 0010	T	01 0011
C	11 0011	L	10 0011	U	01 0100
D	11 0100	M	10 0100	V	01 0101
	etc.		etc.		etc.

The numeric codes shown previously can also be written in a six-position form, like the alphabetic codes, and still be kept consistent by letting the "high-level" binary code be 00. For example:

1	00 0001
2	00 0010
3	00 0011
4	00 0100
	etc.

An interesting predicament occurs when one considers that it is necessary to have a code for a blank space (such as between words) as well as for zero. How can one distinguish between "blank" and "zero"? A convention that has been adopted is to let the blank be represented by 00 0000 and the zero by 00 1010. Notice that the four right-hand positions for this zero are the binary equivalent of the decimal 10. If the BCD zero is used in the above examples, they will not work, since in *arithmetic* operations zero must be represented by 00 0000, not 00 1010. In *nonarithmetic* operations, such as printing a telephone book, the 00 1010 kind of zero can be used. To overcome this difficulty, the programmer must keep his BCD arithmetic separate from his BCD nonarithmetic, and be sure, for his arithmetic operations, to convert the 00 1010 to 00 0000. When it comes to printing the results, he converts the zeros back to 00 1010's to distinguish between zeros and blanks.

There are also BCD codes that correspond to special characters such as commas, periods, etc. For example, in Fig. 2-16, the card code for a comma is 0-3-8. Its BCD code is 01 1011. In like manner, the card code for a period is Y-3-8, and its BCD code is 11 1011. Notice that the four right-hand positions represent the binary quantity corresponding to the sum of the decimal numeric values in the card, the $8 + 3$ in this case.

EXERCISES

1. Each digit of a BCD numeral is represented by a four-place binary numeral. What is the largest binary numeral required to represent a single-decimal digit?
2. What is the largest value that can be represented by a four-place binary numeral?
3. Is the BCD efficient in its use of the binary positions? Does it waste some positions? How much is lost?
4. How does BCD compare with standard binary notation, as far as convenience is concerned, in translation to and from decimal notation?

Part III. Floating Point

Numbers are represented in computers in binary, decimal, or other schemes of notation. Regardless of the method, there is often a great deal of difficulty in planning an extended fixed point calculation. This difficulty lies in the fact that it is hard to foresee the results of hundreds or thousands of arithmetic steps and to allow in advance for the necessary contingencies associated with keeping track of the decimal point. For illustration, assume that the computer stores numerals of ten decimal digits and sign. The numeral 0002905768 can be stored in the computer. Since most digital computers do not have a built-in decimal point, this numeral can represent various magnitudes, depending on where the decimal point is intended to be. The caret ($\wedge$) is used below to show the intended location of the point; thus, the numeral can represent the integer 2905768 $\wedge$ or it can represent the decimal number 29 $\wedge$ 05768.

It is necessary for the programmer to keep track of the decimal point; this makes the task of coding complicated. In order to add $29 \wedge 05768$ to $30002582 \wedge 53$, the decimal points must be aligned by shifting. Computers have operational codes that can be used to shift or move the contents of the arithmetic unit left or right for this alignment process. If these two numbers are in storage and are to be added, the larger one in this example will first have to be placed in the arithmetic unit and then, with a separate instruction, shifted three places to the left before the second number can be added.

$$
\begin{array}{r}
30002582 \wedge 53 \\
29 \wedge 05768 \\
\hline
30002611 \wedge 58768
\end{array}
$$

But now the sum has more digits than can be stored in one location, so the sum must either be rounded to $30002611 \wedge 59$ or truncated to $30002611 \wedge 58$. As only ten digits are retained, it is vitally important to keep the ten most significant digits; notice the magnitude of error introduced if $02611 \wedge 58768$ is retained in storage and the 3, which is 3×10^7, is "lost." If the characteristics of a calculation are unknown, there is always a possibility of accumulating, without realizing it, a sum or a product of more than ten digits; if the most significant digits are "lost", the calculation is worthless.

In addition to these alignment problems, fixed point is also subject to another limitation. For the computer described here, the largest integer that can easily be represented is $9999999999 \wedge$ and the smallest, of course, is $-\wedge 9999999999 \wedge$. Various fractions can be represented also, as $3 \wedge 005632027$ and $-\wedge 0000000001$. In scientific and engineering computation, values such as 9.71×10^{17}, 0.365×10^{-9}, and 2.105×10^{11} are frequently needed, and the problem of decimal point location, or scaling, may become extremely difficult. These problems lead to the development of another system of notation called *floating point*.

Floating-point is a scheme similar to the familiar system called *scientific notation*. Here we will take two of the digits and use them as integer exponents of 10. This means that there are now only eight digits available to represent the significant part of a number, but for most data derived from measurements this is entirely adequate. Now the numeral 3000258253 is interpreted as $\wedge 30002582 \times 10^3$. The number is expressed with the assumed decimal point on the left followed by eight digits. The two rightmost digits are the power of 10 which will correctly position the point. Because the power of 10 can be positive or negative, and it is convenient to represent this by two digits without sign, the convention used is

Exponent Digits	Meaning
53	10^3
52	10^2
51	10^1
50	10^0
49	10^{-1}
48	10^{-2}.

The leftmost digit is always significant, so the number is always expressed as a fraction $0.1 < n \leqslant 1.0$ times some integral power of ten. This is illustrated in the table (p. 152). Notice that the floating point representation for zero itself falls outside of this range. However, zero can be represented by a very small number such as $0.10000000 \times 10^{-50}$, or 0.1000000000.

Standard Decimal Notation	Scientific Notation[1]	Machine Notation
26.1	0.261×10^{2}	2610000052
−0.000000057	-0.57×10^{-7}	−5700000043
−3.1415	-0.31415×10^{1}	−3141500051
6563210.887	$0.6563210887 \times 10^{7}$	6563210957

[1] An alternative notation is shown in the Glossary (Appendix D) under "floating point calculation."

Notice that in the last example there are ten digits in the decimal notation but this is rounded to eight digits for machine use. Some computers are constructed to operate in floating point. That is, their electronic circuitry automatically performs the necessary shifting before addition etc. Other computers do not have this "built-in" facility, and must be programmed especially to do this. In this latter case, the regular user of the computer is not involved, since the compiler, discussed in Chapter 2, would have been developed to do the necessary shifting, etc., for him automatically.

There are programming techniques called "multiprecision" that permit use of more significant digits than the machine is designed for. For example, in double precision, each number may be assigned two locations so that the sum or product could be twice as large as normally accommodated.

The use of floating point notation thus makes the machine notation more like that used by scientists and engineers and relieves the programmer of many difficult coding tasks involving scaling of calculations. By floating point notation the capacity of a storage location has been extended so that a value of 9999999999, meaning $0.99999999 \times 10^{49}$, can be stored. Likewise a value of −1000000000, meaning $-0.10000000 \times 10^{-50}$, can be stored. With this range, most scientific computation can be accommodated.

Although floating point arithmetic can relieve a computer user of much labor in the planning of his problem, it introduces a difficulty as well: It can develop a false sense of security. In other words, just because it is easy to retain eight significant digits in a floating point calculation does not mean that the digits are necessarily "good." As suggested in the discussion of approximation in Chapter 3, "results are no better than the numbers that produced them." For example, in the calculation

$$A = \pi R^2,$$

if

$$R = 3.756 \text{ and } \pi \text{ is used as } 3.14,$$
$$R^2 = 14.107536$$

and

$$A = 3.14(14.107536) = 44.29766304.$$

However, there is some question about the "real" accuracy of these figures beyond the second decimal place simply because π itself was expressed with but two figures to the right of the decimal point. In floating point notation, however, the R^2 would be retained as 1410753652 and A would be retained as 4429766352. The "7663" part of A would be "carried along" in subsequent calculations but is of questionable value. The programmer must still be concerned whether the result of floating point calculations is "correct," that is, whether or not the error introduced by round-off and approximations is within some specified limit.

EXERCISES

1. Express in floating point notation: (a) 1086; (b) 0.025901; (c) 87.6747; (d) 6×10^{23}.

2. Write the following floating point numerals in decimal notation: (a) 1234567850; (b) 6300000045; (c) 2820511354; (d) 0.1000000062.

3. Write the factors and the product in floating point notation: $31785021 \times 71864214 = 2284205551138494$.

4. In (3), what is the error that is introduced by expressing the product in floating point, using the methods of (a) truncation and (b) rounding (if eight significant digits are retained)?

Appendix B

RELATED TOPICS

The purpose of Appendix B is to bring together certain topics that are related to the material in the body of this book but that are outside the scope of the four chapters.

A Square Root Method

In Chapter 3 a method for the extraction of square root was described. There are several possible methods, and the following shows another approach to the problem. It is based on an interesting but more complicated method than that given in Chapter 3. The method is often used by hand-computer operators but not with large digital computers. It is a demonstration of an iterative process, and one that introduces some interesting numerical relationships.

Table I lists x, x^2, and the differences between successive values of x^2 as x takes on successive positive integral values.

TABLE I

x	x^2	Difference
0	0	
		1
1	1	
		3
2	4	
		5
3	9	
		7
4	16	
		9
5	25	
		11
6	36	

Note that the differences form a series of odd numbers starting with 1. Also note that, if the first two differences are added, their sum is equal to 2^2; if the first three are added, their sum is equal to 3^2; and so on for additional combinations. In other words, the sum of the first x differences is equal to x^2.

If this procedure is reversed, that is, if one subtracts the odd series, one-by-one, from a number, y, the *number* of subtractions required to reduce y to zero is the square root of the number.

The following example demonstrates the method with a perfect square.

PROBLEM 1. Obtain the square root of 25.

$$\begin{array}{rl} 25 & \\ \underline{-1} & \text{1st subtraction} \\ 24 & \\ \underline{-3} & \text{2nd subtraction} \\ 21 & \\ \underline{-5} & \text{3rd subtraction} \\ 16 & \\ \underline{-7} & \text{4th subtraction} \\ 9 & \\ \underline{-9} & \text{5th subtraction} \\ 0 & \end{array}$$

That is, 5 is the square root of 25, as demonstrated by the process, since five subtractions reduced the original quantity to zero.

The following example, similar to the one above, will point out the necessity for modification of the procedure when roots for other than perfect squares are sought.

PROBLEM 2. Obtain the square root of 29.

$$\begin{array}{rl} 29 & \\ \underline{-1} & \text{1st subtraction} \\ 28 & \\ \underline{-3} & \text{2nd subtraction} \\ 25 & \\ \underline{-5} & \text{3rd subtraction} \\ 20 & \\ \underline{-7} & \text{4th subtraction} \\ 13 & \\ \underline{-9} & \text{5th subtraction} \\ 4 & \\ \underline{-11} & ? \end{array}$$

We know that the root is not 5, but we also know that the usual subtraction process cannot be continued since the next number to be subtracted, 11, is larger than 4 and would not reduce the quantity to zero. What to do?

The following paragraphs develop the necessary modifications to the "perfect square" procedure so that it can also be applied to numbers which are not perfect squares.

An examination of Table I reveals a relationship between the x column and the Difference column. This relationship is that, for a given x, the "next" difference, N, can be represented by the expression

$$(2x + 1).$$

For example, the next difference beyond $x = 4$ is equal to

$$(2x + 1) \text{ or } (2 \times 4) + 1 = 9.$$

Likewise, the next difference for $x = 26$ would be

$$(2 \times 26) + 1 = 53.$$

This relationship allows one to extend the original table from any arbitrary point, without starting at the beginning; for example:

x	x^2	Difference
25	625	
		51
26	676	
		53
27	729	

The reason for wanting to be able to do this is that when the subtraction process has reached the point where the next subtraction is not possible without producing a negative result, as in the example for the square root of 29, it is necessary to interrupt the procedure, make a modification, and continue in the same manner as before.

The modification is essentially a matter of shifting the subtraction process to the right and continuing, but starting at a new location in the series of odd numbers. Multiplication or division of a number by 10 has the effect of shifting the decimal point one place to the right or left, respectively; and the terms "shift right" and "shift left" are frequently used as shorthand expressions for multiplication and division, respectively. Notice that shifting the decimal point in one direction is equivalent to shifting the number in the other direction.

In operating with squares and square roots it is understood that if a number is multiplied by 10, its square is multiplied by 100. This is another way of saying that if one knows the square root of a number, he can obtain the square root of 100 times the number merely by multiplying the square root of the number itself by 10. For example, since the square root of 25 is 5, the square root of 25×100 is equal to 5×10, or the square root of 2500 is equal to 50.

To demonstrate this property using the square root method under discussion, when a shift is required, the following examples are given to obtain the square root of 4 and 400.

PROBLEM 3. Obtain the square root of 4.

$$\begin{array}{rl} 4 & \\ -1 & \text{1st subtraction} \\ \hline 3 & \\ -3 & \text{2nd subtraction} \\ \hline 0 & \end{array}$$

In this case two subtractions reduced the original number to zero; according to this method, then, the square root is 2.

To demonstrate the method when the shifting process is introduced, start as before.

$$\begin{array}{rl} 4 & \\ -1 & \text{1st subtraction} \\ \hline 3 & \end{array}$$

This time, however, shift and start at a new place in the series of odd numbers. Consider that by the shifting process the 3 remainder, above, has been changed to 300 by shifting two places to the right. Recall that a multiplication of a number by 100, which is the same as a shift of two places to the right, introduces a change in that number's square root by a multiplication by 10. The starting value of the series shown below, 21, will be explained later. The following series of subtractions shows that, if one starts subtracting with 21, the value 300 will be reduced to zero in ten steps.

	Step
300	
−21	1
279	
−23	2
256	
−25	3
231	
−27	4
204	
−29	5
175	
−31	6
144	
−33	7
111	
−35	8
76	
−37	9
39	
−39	10
0	

The previous single subtraction of 1, which reduced the original 4 to 3, is equivalent to ten subtractions, had they started with 400 instead of 4, as follows.

	Step
400	
−1	1
399	
−3	2
396	
−5	3
391	
−7	4
384	
−9	5
375	
−11	6
364	
−13	7
351	
−15	8
336	
−17	9
319	
−19	10
300	

The above illustrates that, after a shift, the next number to subtract is, in this case, 21. In general, this can be expressed as

$$N = (2S + 1)$$

where S is equivalent to the number of times that subtraction has taken place before the shift. That is, after ten subtractions (equivalent to one subtraction before a shift) the next number to subtract is

$$N = (2S + 1) \text{ or } (2 \times 10) + 1 = 21.$$

In Problem 2, to obtain the square root of 29, after five subtractions, a shift was called for. According to the above, the next number to subtract after the shift would be

$$N = (2S + 1) \text{ or } (2 \times 50) + 1 = 101,$$

noting that the five subtractions before the shift are equivalent to fifty subtractions after the shift.

The next number to subtract, N, after a shift can also be obtained if one knows the *last number subtracted* before the shift, L. Note that this is distinct from the number of *times* subtractions have been performed, as in the relationship just described.

This new relationship, based on the number subtracted, is

$$N = 10(L + 1) + 1.$$

In the case of the square root of 29,

$$N = 10(9 + 1) + 1 = 101,$$

and for the other example,

$$N = 10(1 + 1) + 1 = 21.$$

The example given above (extract the square root of 29) is worked out below to three decimal places. Notice that as additional shifts are called for, the above formula for N is applied again.

```
   29|
   -1|
   28|
   -3|
   25|
   -5|          5 subtractions
   20|
   -7|
   13|
   -9|
   400| <---- Shift here; N = 10(9 + 1) + 1 = 101
  -101|
   299|
  -103|         3 subtractions
   196|
  -105|
   9100| <--- Shift here; N = 10(105 + 1) + 1 = 1061
  -1061|
   8039|
  -1063|
   6976|
  -1065|
   5911|
  -1067|        8 subtractions
   4844|
  -1069|
   3775|
  -1071|
   2704|
  -1073|
   1631|
  -1075|
   55600| <- Shift here; N = 10(1075 + 1) + 1 = 10761
  -10761|
   44839|
  -10763|
   34076|
  -10765|       5 subtractions
   23311|
  -10767|
   12544|
  -10769|
    1775
```

The square root of 29, computed this far, is 5.385.

It is advantageous to use the shifting process whenever possible; that is, to extract the square root of 1234, one might start in either of two ways, as follows:

EXAMPLE.

```
 1234         1234|
  −1          −1  |
 ----         ----|
 1233         1134|
  −3          −3  |          3 subtractions
 ----         ----|
 1230   or     834|
  −5          −5  |
 ----         ----|
 1225          334|  ←——— Shift. N = 10(5 + 1) + 1 = 61
 etc.          −61|
               273|
               −63|
               210|
               −65|          5 subtractions
               145|
               −67|
                78|
               −69|
                 900|  ←——— Shift. N = 10(69 + 1) + 1 = 701
                −701|        1 subtraction
                  19900|  ←—— Shift. N = 10(701 + 1) + 1 = 7021
                  −7021|      2 subtractions
                  12879|
                  −7023|
                    585600|  ← Shift. N = 10(7023 + 1) + 1 = 70241
                    −70241|
                    515359|
                    −70243|
                    445116|
                    −70245|
                    374871|
                    −70247|      8 subtractions
                    304624|
                    −70249|
                    234375|
                    −70251|
                    164124|
                    −70253|
                     93871|
                    −70255|
                     23616
```

The square root computed thus far, then, is 35.128. By a continuation of the process any desired degree of accuracy can be obtained.

For any number, the starting procedure is as follows: Divide the number into groups of two digits each way from the decimal point, and start the subtraction process under the right-hand digit of the left-most pair as shown (p. 161). The arrows indicate the position at which the subtraction process should start; that is, subtract 1.

$$\underbrace{6\;7}\;\;\underbrace{0\;1}.$$
$$\uparrow$$
$$\underbrace{4}\;\;\underbrace{6\;3}\;\;\underbrace{2\;7}.$$
$$\uparrow$$

$$.\underbrace{1\;3}\;\;\underbrace{6\;7}$$
$$\uparrow$$
$$.\underbrace{0\;7}\;\;\underbrace{6\;2}.$$
$$\uparrow$$

An interesting point in considering this method is the manner in which the process converges toward the root. The process approaches the root from the "low" side; that is, at any given step in the process the approximation to the root at that point is equal to or less than the root. It is easy to see why this is so, since the subtraction process, in reducing the number, always selects a value which is equal to or less than the difference. (Otherwise the result is negative.) Therefore, at any stage the sum of the "contributions" to the root is equal to or less than the root. Notice how in the example above the different estimates, called E below, converge to obtain the square root of 12.34.

E	E^2	$(12.34 - E^2)$
(3.0)	9.0	3.34
(3.5)	12.25	0.09
(3.51)	12.3201	0.0199
(3.512)	12.334144	0.005856
(3.5128)	12.33976384	0.00023616

This method is well suited to hand operations with desk calculators, but would probably not be used with large digital computers, since certain other methods, such as the method discussed previously, in Chapter 3, are superior.

A Cube Root Method

The following discussion is primarily for the individual who likes to study numerical relationships and to see what he can discover about these relationships. The procedure developed is a repetitive one and is included for two reasons: (1) because it is an interesting extension of the square root method previously described; and (2) because, in the presentation, the study of the relationships between sets of numbers by the use of differences is demonstrated. In numerical analysis it is often advantageous to consider the treatment of differences in order to plan the solution of problems. This example, like the square root example just described is intended to demonstrate an iterative process rather than to develop or recommend a mathematical method.

Table II is similar to the one prepared for square root. The method is of the same type as that used for square root in that successive subtractions are performed using the first differences of x^3. This corresponds to the subtraction of the series of odd numbers in the square root procedure. Note that the sum of the first two values in the column headed "First Differences of X^3" equals 2^3; the sum of the first three values equals 3^3; the sum of the first four values equals 4^3, etc.

The other columns of differences are included (1) to illustrate further some of the points developed in Chapter 3 on the usefulness of examining differences of successive values of a function as one studies the function; and (2) to aid in the development of this method of extraction of cube root.

TABLE II

X	X^3	First Differences of X^3	Second Differences of X^3	Third Differences of X^3	Fourth Differences of X^3
0	0				
		1			
1	1		6		
		7		6	
2	8		12		0
		19		6	
3	27		18		0
		37		6	
4	64		24		0
		61		6	
5	125		30		0
		91		6	
6	216		36		0
		127		6	
7	343		42		0
		169		6	
8	512		48		0
		217		6	
9	729		54		0
		271		$\square$	
10	1000		$\square$		
		$\square$			
11	$\square$				
12					

First, Table II illustrates that, for an expression of degree n, the nth differences are constant and the $(n + 1)$th differences are zero.[1] It is also useful to notice that in starting with the right-hand column of differences (in this case the fourth differences), it is possible to work backwards and determine the value of the

[1] It is useful to consider different ways of constructing columns of differences. For example, the second difference can be constructed by considering successive values of the function in groups of three. If these are called a, b, and c, the first differences are $(b - a)$ and $(c - b)$. Also the second difference would be $(c - b) - (b - a) = c - 2b + a$. In other words, the second difference that results from any three consecutive values is equal to the sum of the first and third values less twice the second value. This kind of procedure may be more convenient to program for a computer than the "by hand" method. Similar relationships for higher differences can also be developed.

function. This is shown in the Table II by the use of arrows. Quantities are to be added along the diagonal arrows and the sum written at the head of the vertical arrows. For example, start at the fourth difference (which is zero) opposite the value of $x = 7: 0 + 6 = 6,\ 6 + 42 = 48,\ 48 + 169 = 217,\ 217 + 512 = 729$. In like manner the empty boxes can be filled in; and, this having been done, starting again at the fourth difference column extend the table as far as desired.

Second, the series to subtract for cube root is more complicated than that used in the square root example. However the two procedures have the same basic requirement: after a given subtraction, determine the next number to be subtracted.

For any part of the process where the values to be subtracted are in regular sequence, it can be seen, by examination of Table II, that the next value to subtract, N, can be determined if one knows the last value just subtracted, L, and the number of times subtractions have been performed to this point, P. This relationship is

$$N = (L + 6P).$$

For example, if $P = 9$ and $L = 217$,

$$\begin{aligned} N &= 217 + (6 \times 9) \\ &= 217 + 54 \\ &= 271 \end{aligned}$$

which is seen to be the next entry in the table.

It would be more convenient however to be able to determine the value of N solely upon the basis of the number of subtractions performed. This convenience is to be noted especially when what was referred to in the square root procedure as "shifting" is required.

An examination of Table II will show that it is possible to develop the following relationship to give N for any P,

$$N = 3P\,(P + 1) + 1,$$

or, for the above example, with $P = 9$,

$$\begin{aligned} N &= (3 \times 9)\,(9 + 1) + 1 \\ &= (27)\,(10) + 1 \\ &= 271. \end{aligned}$$

The cube root process calls for the grouping of the number into groups of 3, whereas the square root process called for groups of 2. The following examples show the process for extracting the cube root.

EXAMPLE. Find the cube root of 8:

$$
\begin{array}{r|l}
8 & \text{1 subtraction (equivalent of 10 for getting } N \text{ after shift)} \\
\underline{-1} & \\
7000 & \leftarrow \text{Shift.} \quad N = (3 \times 10)\,(10 + 1) + 1 = 331 \\
\underline{-\ 331} & \\
6669 & \\
\underline{-\ 397} & \longrightarrow \ N = (3 \times 11)\,(11 + 1) + 1 = 397 \\
6272 & \\
\underline{-\ 469} & \longrightarrow \ N = (3 \times 12)\,(12 + 1) + 1 = 469 \\
5803 & \\
\underline{-\ 547} & \qquad \text{etc.} \\
5256 & \\
\underline{-\ 631} & \\
4625 & \\
\underline{-\ 721} & \quad \text{10 subtractions} \\
3904 & \\
\underline{-\ 817} & \\
3087 & \\
\underline{-\ 919} & \\
2168 & \\
\underline{-1027} & \\
1141 & \\
\underline{-1141} & \\
0 &
\end{array}
$$

This, then, is the equivalent of 20 subtractions, and the 20 may be considered as

$$20 = \sqrt[3]{8000}$$

or

$$2.0 = \sqrt[3]{8.000}.$$

Without shifting, this example would have been as follows:

$$
\begin{array}{rl}
8 & \\
\underline{-1} & \text{1st subtraction} \\
7 & \\
\underline{-7} & \text{2nd subtraction.} \\
0 &
\end{array}
$$

EXAMPLE. Find the cube root of 1906.624:

$$
\begin{array}{rl}
1906.624 & \leftarrow \text{1 subtraction (equivalent of 10)} \\
\underline{-1} & \\
906\ 624 & \leftarrow \text{Shift. } N = (3 \times 10)\,(10 + 1) + 1 = 331 \\
\underline{-331} & \\
575\ 624 & \quad \text{2 subtractions} \\
\underline{-397} & \\
178\ 624 & \leftarrow \text{Shift. (The next value to subtract would be 469 from 178.) For the first} \\
\underline{-43\ 561} & \quad \text{subtraction after the 2nd shift, the original 1 (which had been made the} \\
135\ 063 & \quad \text{equivalent of 10) is now the equivalent of 100, and the 2 subtractions become} \\
\underline{-44\ 287} & \quad \text{the equivalent of 20, or 120 total; therefore } N = (3 \times 120)\,(120 + 1) + 1 = \\
90\ 776 & \quad 43561. \\
\underline{-45\ 019} & \\
45\ 757 & \\
\underline{-45\ 757} & \quad \text{4 subtractions.} \\
0 &
\end{array}
$$

The result then is 12.4, or if the number had been 1.906624, the result would have been 1.24; if the number had been 1906624, the result would have been 124.

As can be seen, this process is complicated and, therefore, not too practical as a method to extract cube root. It does, however, point out certain numerical relationships and shows that much can be learned about mathematical expressions by the examination of successive differences of a series of values of the expressions.

More Numerical Approximations

In Chapter 3, numerical approximations were discussed and some examples were given. A related technique is that in which one expression is used to approximate another more complicated expression.

A digital computer can perform a large amount of arithmetic in a short time but it is generally desirable to reduce operations to the simplest form if for no other reason than it takes less time to complete a problem if simple and rapid procedures are used.

The examples below demonstrate some techniques that are useful in determining expressions that can be used to approximate other expressions.

The relationship between two well-known temperature scales, Centigrade and Fahrenheit, is expressed as follows:

$$C = \frac{(F - 32)5}{9}.$$

This means that for any particular value of F a corresponding value of C can be located by a substitution and some simple arithmetic. This is a linear relationship since the independent variable, F, is of the first power. That is to say, if this relationship is reduced to a graph it will be a straight line. The graph in Fig. B-1 (p. 166) shows this relationship.

Only a few values were used to construct the graph, and actually only two were required, since if the original relationship holds, a straight line connecting any two points and extended in each direction will include all points represented by the relationship.

In using this graph as an aid to calculation then, to get the value of C for any F, all that is required is to locate the desired point along the F-scale and erect a perpendicular to the F-axis through the point of interest. At the point where this perpendicular intersects the straight line, erect another perpendicular to the C-axis and read on the C-axis where this perpendicular intersects it.

There are limitations, of course, upon the accuracy of graphical methods since drawing graphs and reading graphs can only be done to a comparatively low level of accuracy, unless special procedures, which may be time-consuming, are used.

The obvious thing to do, of course, if accuracy is desired, is to make a substitution in the formula given above. It is quite likely that this is what would be done if one is using a digital computer for a problem which requires conversions between Centigrade and Fahrenheit.

In summary then, if a relationship between two variables is a linear one (can be represented by a straight line), it is usually convenient to "substitute in the formula."

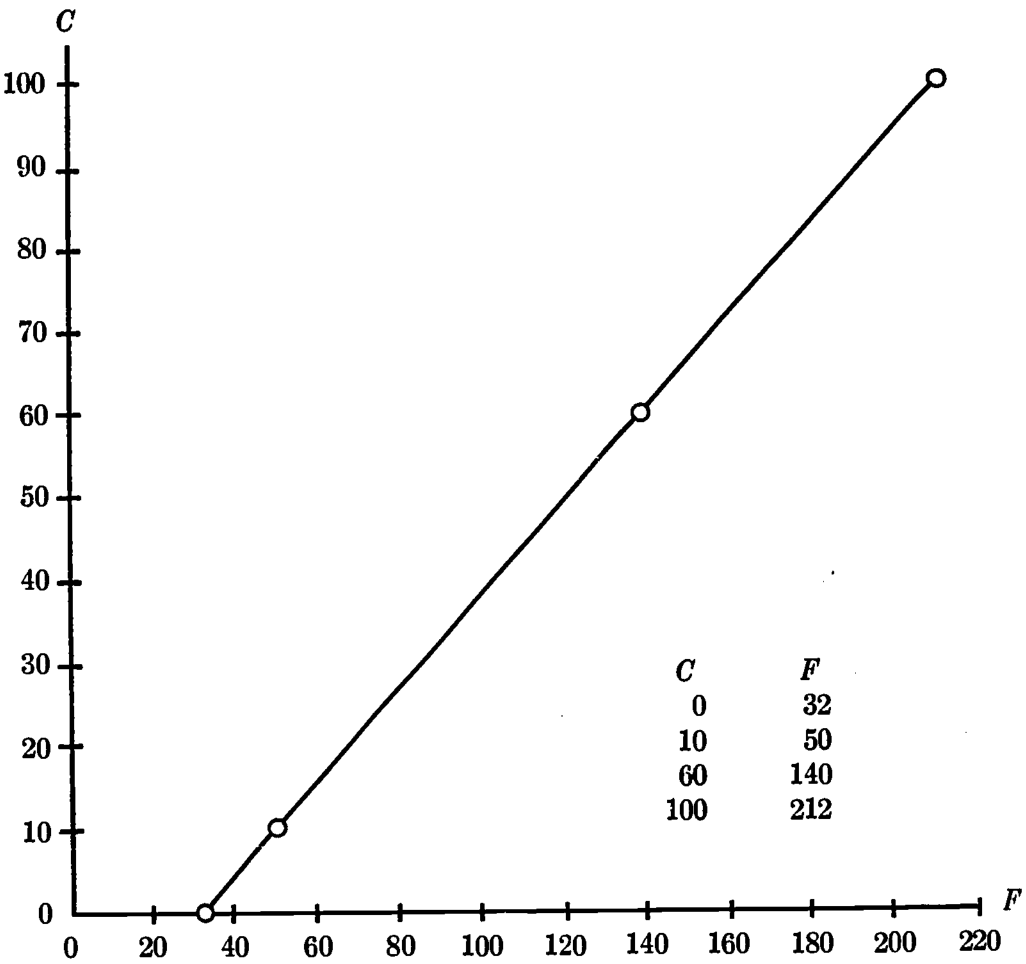


Fig. B-1. Relation between Fahrenheit and Centigrade Scales

Many relationships, however, are not linear, so they cannot be represented by a straight line. One of the most common, of course, is that for square root. This can be expressed as follows:

$$y = x^2.$$

Figure B-2 shows that this relationship is not a straight line. This is indicated by the fact that the independent variable, x, is not to the first power, but to the second.

With this graph it is possible, using a technique similar to that desciibed with the previous linear graph, to select any value of y, and the corresponding value of x will be the square root of y.

The problem of inaccuracy due to the construction and reading of the graph is similar to that in the linear case.

Curves that represent other relationships can also be treated in the same way, such as those including higher powers than the second: ($y = x^3$). They all have limitations on accuracy.

Sometimes the "formula" approach is not convenient. For example, the formula in the square root curve is the square root process itself which is considerably more trouble to carry through than the formula relating C and F in the first example.

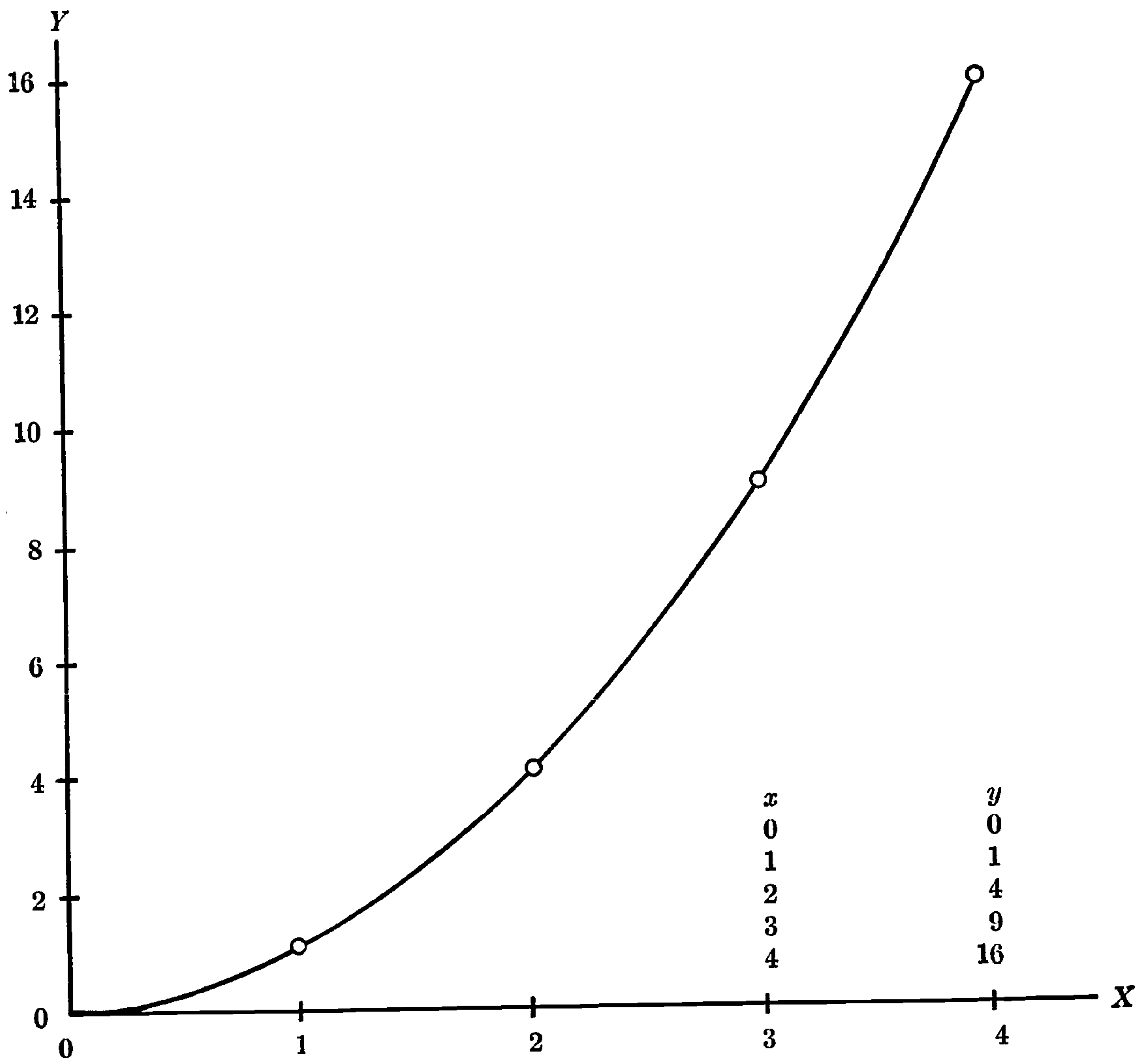
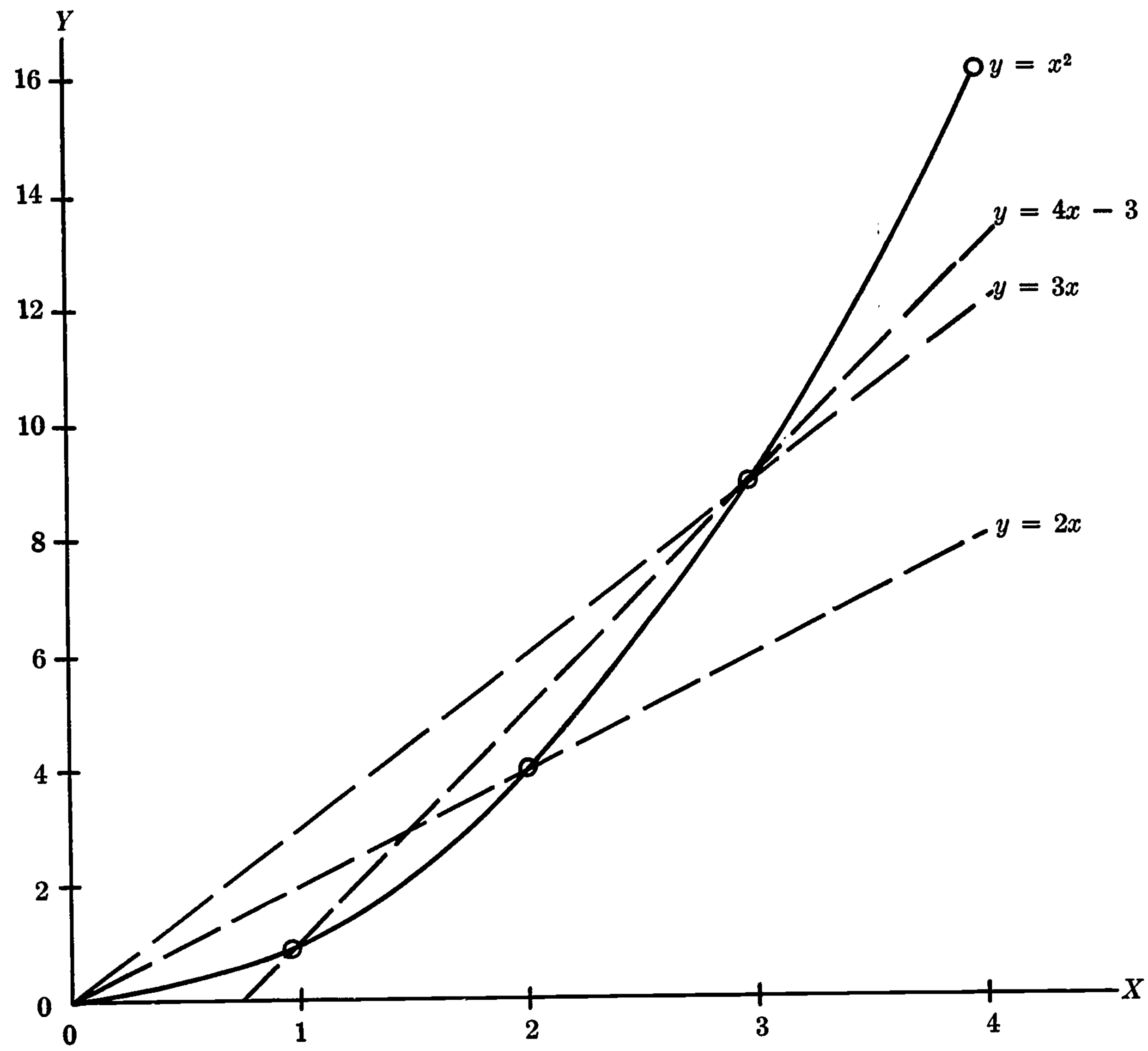


Fig. B-2. Graph of $y = x^2$

The point of the above comparison was to suggest that it would be nice if a second-degree relationship, or curve, such as $y = x^2$, could be represented by a first-degree or straight-line relationship such as $y = x$.

Within limits, this can be done, and the limits are imposed by the accuracy required. Another way of saying this is as follows: Let a straight line represent a curved line provided it is not "too much" different. The diagram in Fig. B-3 demonstrates this point. It is only a reproduction of the previous graph with certain straight lines connecting certain points.

Notice that if the curve $y = x^2$ had been replaced by one of the other linear expressions, as an approximation to it, certainly the arithmetic in going from one to the other would be simpler but error would also be introduced. Also notice that the amount of error is related to "how far the curve is away from the straight line." For example, the curve $y = 3x$, as an approximation for $y = x^2$, is excellent at the point $x = 3$. In the same way the curve $y = 4x - 3$ is an excellent approximation for $y = x^2$ at $x = 3$. This is to say, at the place where one curve crosses another "all is well." It's "out in the middle" where the approximations of curves by straight lines introduce error.



$y = x^2$		$y = 4x - 3$		$y = 3x$		$y = 2x$	
x	y	x	y	x	y	x	y
0	0	0	−3	0	0	0	0
1	1	1	1	1	3	1	2
2	4	2	5	2	6	2	4
3	9	3	9	3	9	3	6
4	16	4	13	4	12	4	8

Fig. B-3. Linear Approximations of $y = x^2$

With this in mind, notice that the curve $y = 4x - 3$ is as good as $y = x^2$ at two points: $x = 1$ and $x = 3$. Everywhere else it misses the curve $y = x^2$, but it does not miss it so much as the curve $y = 3x$. The distance between the curve of interest and the curve set to approximate it indicates the level of accuracy to be expected by that particular approximation. This suggests, therefore, that curves such as $y = x^2$ might be approximated by more than one straight line, each one being used for a short distance only, the distance being determined by how far off the straight line is from the curve. This "how far off" is a measure

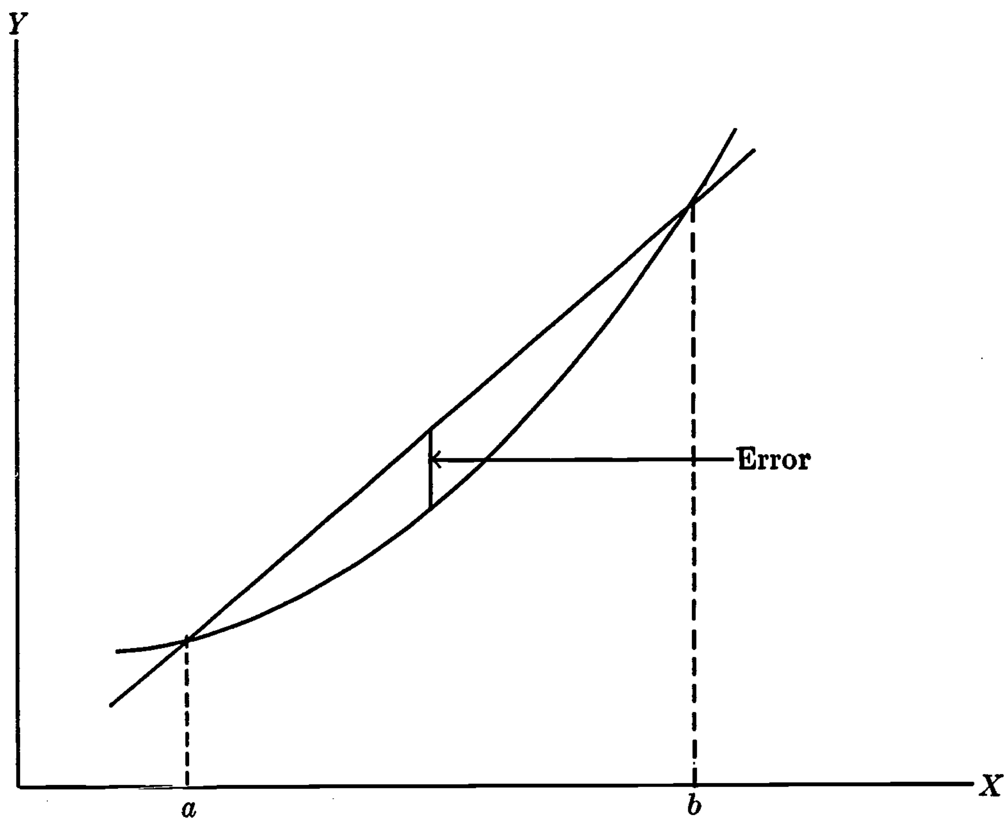


Fig. B-4. Line-Curve Approximation Error

of the error, and since some calculations require greater accuracy than others, different straight lines might be selected. This error is indicated in Fig. B-4.

In other words, in the curve of Fig. B-4, if the error (vertical distance between the curve and straight line) is not too large for the particular application, one might just as well use the straight line (and its simpler arithmetic) for all values of x between the values a and b.

It is easy to imagine, then, a kind of "sliding straight line" being moved along the curve. If the straight line does not fit the curve well enough, that is, make the error-distance small enough, slide it to a new position. For a given level of error, the distance between a and b, which defines the straight line, must get shorter and shorter as the curvature gets sharper and sharper. Likewise, as the curvature gets less and less, and the curve looks more and more like a straight line the distance between a and b can become larger and larger.

Imagine then that a digital computer, or a person for that matter, could extract square root by a method no more complex than that of converting from Fahrenheit to Centigrade if he had the proper set of *linear* formulas corresponding to the straight lines sliding around the curve $y = x^2$. The procedure would simply be (1) determine what the range of interest was (that is, between a and b), (2) select the corresponding linear approximation that will keep the error small enough, and (3) calculate.

There is a very important extension to this whole idea that one curve (a straight line) can be made to approximate another. This extension is that for some curves it may be more desirable to approximate them by other curves rather than by straight lines. This may allow greater range (a to b, above) with less error.

For example, in Fig. B-5, in which the errors are exaggerated, Curve B, which

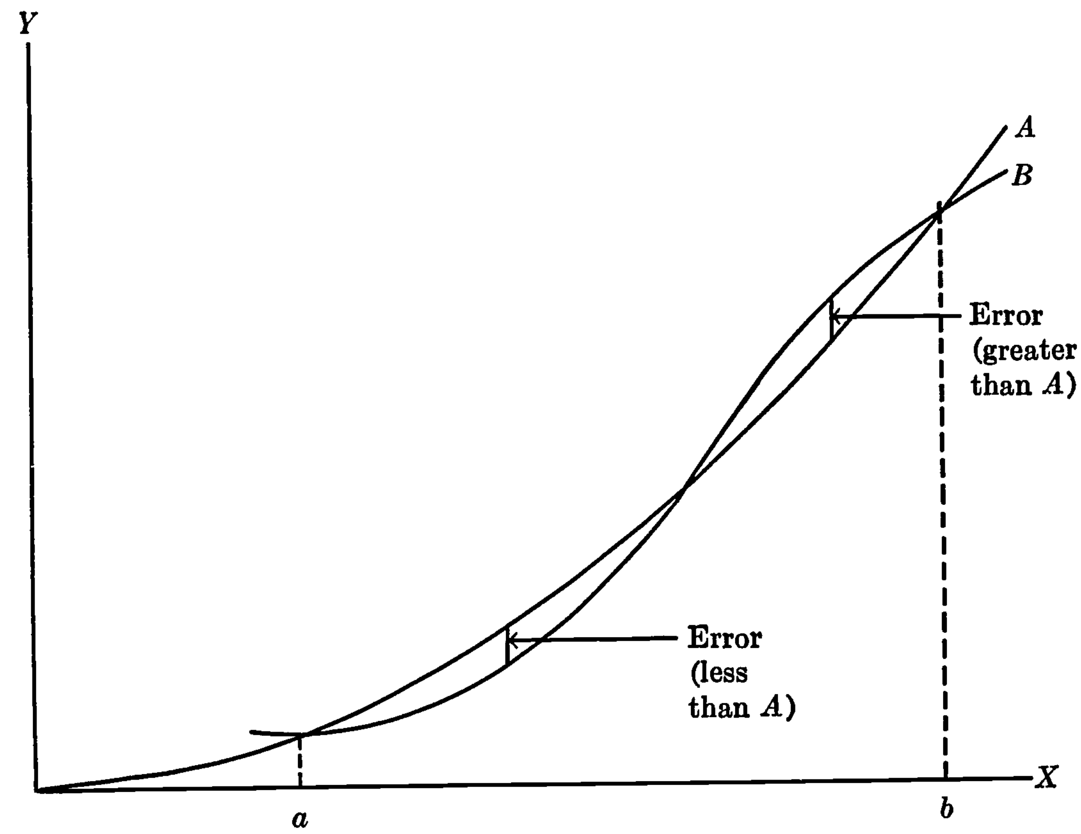


Fig. B-5. Approximating a Curve with Another Curve

is not linear, approximates Curve A with errors that give values that are sometimes less and sometimes greater than the true value. Depending upon the choice of Curve B, the fit can be made better or worse.

The following non-linear expression, which is not actually plotted but merely represented in Fig. B-5 by Curve B, can be used to determine the square root of a number.

$$\sqrt{x} = \frac{.149 + 2.33x + x^2}{1 + 2.33x + .149x^2}$$

This expression, by Hastings, is intended for use in the range 0.1 to 10.0. For values outside of this range the position of the decimal point can be adjusted so that the value falls within the range. For example, the value 75.31 can be considered to be 0.7531 by moving the decimal point *two* places to the left for the calculation. After the calculation the decimal point is moved *one* place to the right in the manner described earlier.

The tabulation (p. 171) gives several values of x in the first column. The second column shows the values of the square root of x as determined by substituting in the above expression. The third column gives the values as obtained from a square root table. The last column gives the error. Notice that the error is sometimes + and sometimes −, indicating that the curve represented by the above non-linear expression oscillates around the curve of the square root function.

Given x	$\sqrt{x}$ Hastings' Approximation	$\sqrt{x}$ From 4-Place Tables	Error
.1	.3175	.3162	+.0013
.5	.7102	.7071	+.0031
1.	1.0000	1.0000	.0000
2.	1.4081	1.4142	−.0061
5.	2.2473	2.2360	+.0113
6.	2.4641	2.4494	+.0147
7.	2.6597	2.6457	+.0140
9.	3.0000	3.0000	.0000
10.	3.1492	3.1622	−.0130
75.31	8.7028	8.6781	+.0247
(.7531)	(.87028)	(.86781)	+(.00247)

Another Sorting Procedure

A method of sorting a series of numbers into an ascending sequence was described in Chapter 4. There are several methods of sorting, of which the following is a second example. This technique, like the one in Chapter 4, and in fact many such systems, is based on successive comparisons of pairs of numbers. To begin, the first two pairs are compared. If they are in order, the second and third, then the third and fourth, then the fourth and fifth, etc., are compared until the list is exhausted. If a particular pair is not in order, these two are reversed and the process is continued. Such a reversal of a pair may disturb an earlier pair so that, after reaching the end of the list, the entire procedure must be done once again from the beginning. Whenever the pair comparisons can be performed, from start to finish, without a single reversal, the set will be in order.

The columns below show the procedure using the numbers given in the previous example. The original list was as follows:

41
18
6
2
19
1
62
110
3
1006.

The first comparison is between 41 and 18. These are not in order, so will be reversed. After this reversal the list is now:

18
41
6
2
19
etc.

Now the second comparison is made between 41 and 6. These are also out of order, so they are reversed, and the list now is

18
6
41
2
19
etc.

At this step 41 is compared with 2 and found to be out of order again. Therefore this pair is reversed, and the list is now:

18
6
2
41
19
etc.

The effect so far is to "drive" the 41 down in the list. Notice that the order of the pairs above each reversed pair may be disturbed.

The lists below show the entire procedure. Numbers with asterisks show reversals.

Original List	Trial No. 1	Trial No. 2	Trial No. 3	Trial No. 4	Trial No. 5	Trial No. 6	Trial No. 7
41	18	6	2	2	1	1	1
18	41*6	18*2	6	6*1	2	2	2
6	41*2	18	18*1	6	6	6*3	3
2	41*19	19*1	18	18	18*3	6	6
19	41*1	19	19	19*3	18	18	18
1	41	41	41*3	19	19	19	19
62	62	62*3	41	41	41	41	41
110	110*3	62	62	62	62	62	62
3	110	110	110	110	110	110	110
1006	1006	1006	1006	1006	1006	1006	1006

Since Trial No. 7 had no reversals, the set is in order.

Binary Arithmetic and Electronics

In Chapter 4 there was a discussion of the process of comparison in certain digital computer applications. In general this was a sequential, two-step process of compare--take action, compare--take action, compare--take action, etc.

There is another important idea in the logic of data handling. This is the concept of what might be called simultaneous comparison or coincidence. The idea depends less on the idea that one thing is smaller or larger than another thing, but more on the idea of whether two or more things exist at all at a given time.

The example on library searching in Example 4 of Chapter 4 made use of this idea. In other words the report was, in a sense, defined or selected or located according to the simultaneous existence of its number on several cards (or tape groups). That is, the "answer" to the searching problem was associated with the locating of a set of conditions which, at a given time, merely existed (not taller or shorter, etc.).

Another example of the utility of the idea of *coincidence*, as distinct from the idea of *comparison*, described in Chapter 4, is given below.

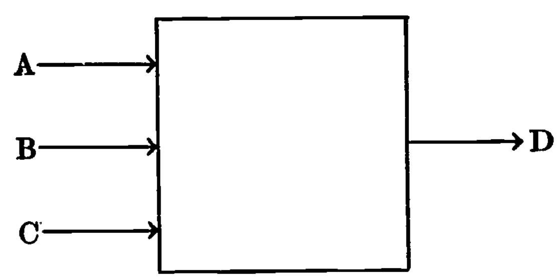
Four situations need to be understood in order to follow the example.

Situation 1. Addition in the binary number system consists of the four following combinations only.

$$\begin{array}{r} 0 \\ +0 \\ \hline 0 \end{array} \quad \begin{array}{r} 0 \\ +1 \\ \hline 1 \end{array} \quad \begin{array}{r} 1 \\ +0 \\ \hline 1 \end{array} \quad \begin{array}{r} 1 \\ +1 \\ \hline (1)\ 0 \end{array}$$

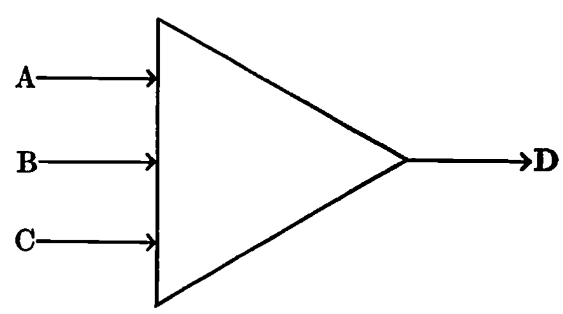
In the right-hand combination, the (1) is "carry-over."

Situation 2. There are electronic devices available that will give out a signal if, and only if, a given number of simultaneous inputs are entered. This can be diagrammed as follows:



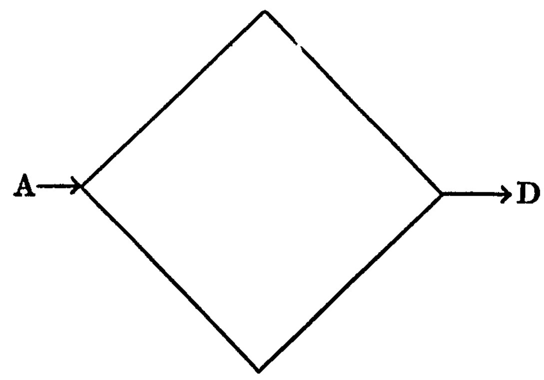
In other words something comes *out* along the D line only if something goes *in* on all *three* of lines A, B, and C. This is called an "AND device," meaning there is output (D) if inputs at A *and* B *and* C exist at the same time.

Situation 3. There are electronic devices available that will give out a signal if any *one* of a given set of inputs is entered. This is diagrammed in much the same way as the AND device above. In this case, however, there will be output at D if A *or* B *or* C provide input. This is called an "OR device," and a different symbol is used.



Situation 4. In addition to AND devices and OR devices, there are also NOT devices (sometimes called INVERTERS). For output, the NOT device "inverts" what it receives as input. That is, for a given operational cycle, if there *is* an input signal at A there *will not* be an output signal at D. Also if there *is not* an

input signal at A, there *will be* an output signal at D.



Situation 1 shows that for the addition of two one-position binary numbers there are only four possibilities. If the numbers have two or more positions, such as

$$\begin{array}{r} 110 \\ +\ 111 \\ \hline 110 \end{array}$$

any column except the right-most may be influenced by the carry-over to it from the column to its right.

Stated another way, in adding two binary numbers, the final answer, or number, in a given *column* is dependent upon three things only:

1. The Top number
2. The Bottom number
3. Was there carry-over from the column to the right?

Suppose the following numbers are to be added.

				C		Carry
	0	0	1	1	1	Top
+	1	1	0	0	1	Bottom
				A		

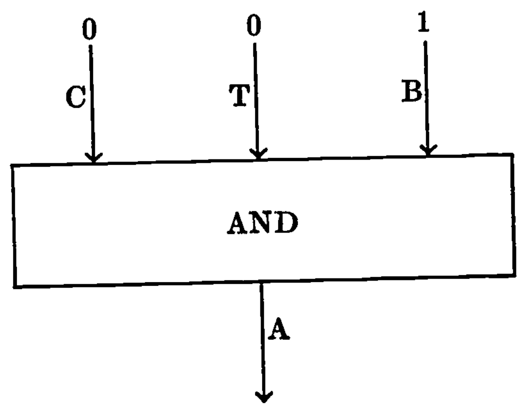
Imagine that the rectangle moves from right to left one position at a time as the adding is done, successively developing the answer one number at a time. Including the possibilities of carry-over, there are then only eight possible *different* conditions to produce a given A in a *given* rectangle. These are the four above *with* carry-over and the four above *without* carry-over:

0	0	0	0	Carry from previous column = zero
0	0	1	1	Top
0	1	0	1	Bottom
0	1	1	0	A

and

1	1	1	1	Carry from previous column = one
0	0	1	1	Top
0	1	0	1	Bottom
1	0	0	1	A.

Note that the above are not additions; they are merely conditions of A (in the rectangle)—the "value" for that column; that is, the carry-overs are not recorded. In a real problem, they would be taken care of when the rectangle "moves left." There are two conditions, that of zero and that of one, which may exist in three places: Carry, Top, and Bottom, or $2^3 = 8$ in all. Of course there are only two types of A: 0 and 1. Imagine that both the 0 and 1 conditions provide short electrical pulses. These pulses may be developed under different circumstances but can make an AND device operate if they are intermixed, thus:



In other words, referring to the diagram above, *something* must go into each of C, T, and B; or, A is not active. That is, in this case, C = 0, T = 0, B = 1. Circuits are so arranged that, as shown in Fig. B-6, if T = 0, impulses enter AND devices 0, 1, 4, 5; and if T = 1, impulses enter AND devices 2, 3, 6, 7. In like manner, for B = 0, impulses enter AND devices 0, 2, 4, 6; and for B = 1, impulses enter AND devices 1, 3, 5, 7. (See footnote 2.)

Fig. B-6 shows how, by proper connections, the *coincidences* of conditions are useful. Suppose we are to add the two numbers below.

	0	0	1	1	1	Top
+	1	1	0	0	1	Bottom

Let it be planned that the devices are connected as shown: the top number is connected to all appropriate T's, and the bottom number is connected to all appropriate B's. (See footnote 2.) As the procedure starts, only the first number is sensed from Top and Bottom; i.e., 1 and 1. Since there is no carry-over this

[2] AND devices can be operated only by ONE impulses. Whenever a diagram calls for a ZERO, such as in Fig. B-6, the necessary ONE will have been previously produced by an INVERTER (not shown). An INVERTER will change a ZERO into a ONE. Accordingly, for the T's, AND devices 2, 3, 6, 7 each receive their impulses "direct", but AND devices 0, 1, 4, 5 each receive their impulses through a separate INVERTER.

NOTE. AND devices 2, 6, 7 will also get T impulses but will not be effective because either B or C or both B and C will not have the proper condition. In like manner, B impulses will also go to AND devices 1, 5, 7, but for like reasons they will not be effective.

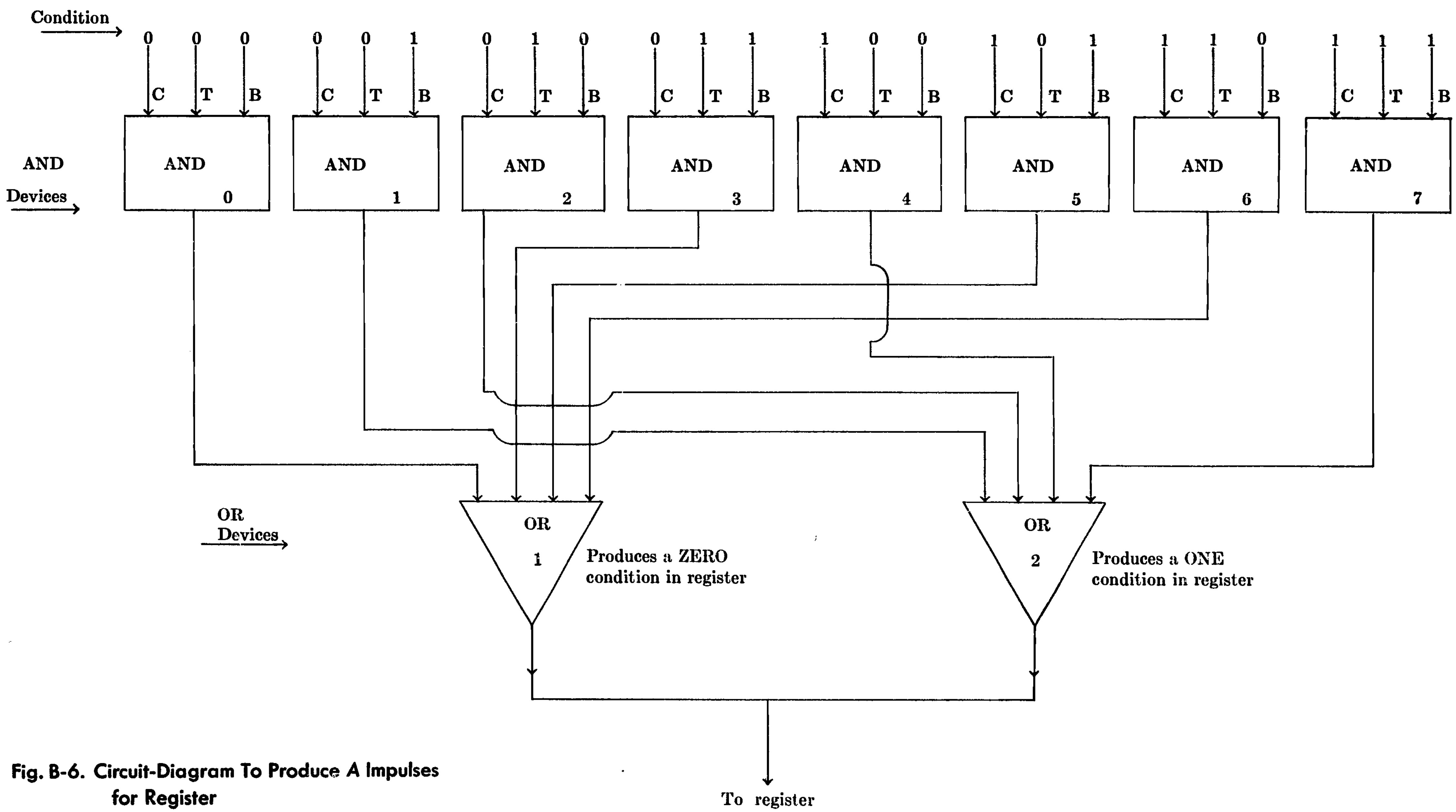


Fig. B-6. Circuit-Diagram To Produce A Impulses for Register

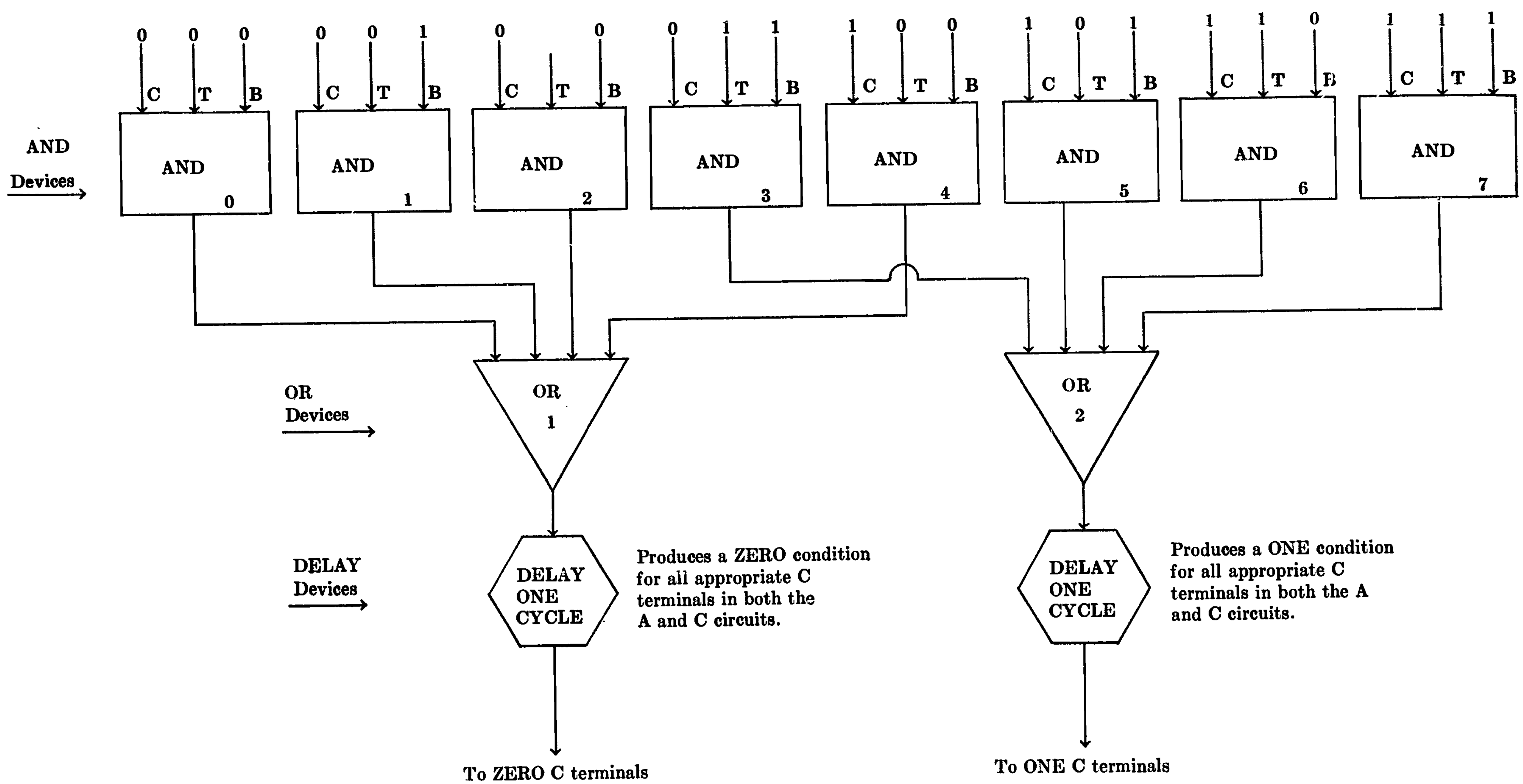


Fig. B-7. Circuit-Diagram To Produce C Impulses

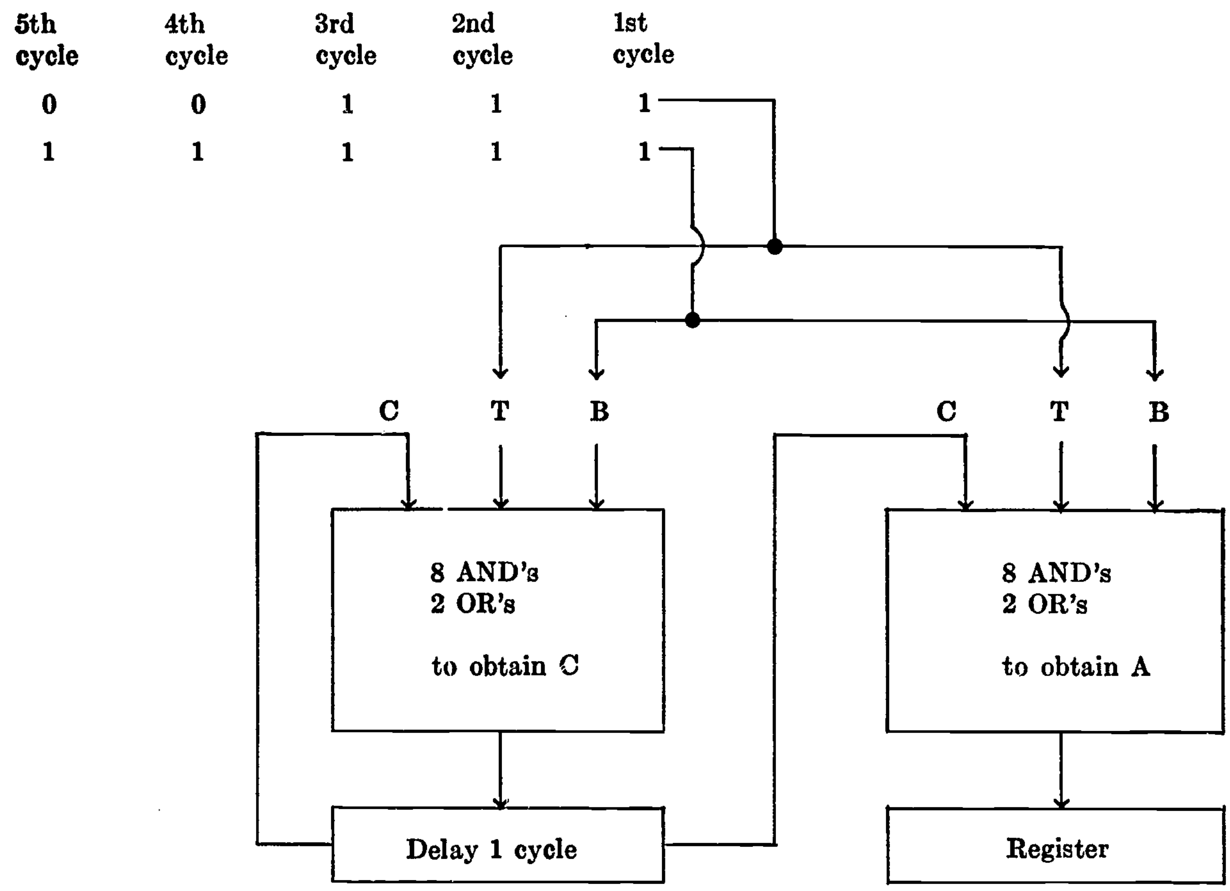


Fig. B-8. A Block Diagram for Binary Addition

time (it being the first), AND device 3 will be made active because its inputs are 0, 1, and 1.

Furthermore, the line from AND device 3 goes to OR device 1 and produces a 0 condition, which in turn is recorded in the first position of a register used to develop the sum. The second positions for T and B are then sensed, etc. To take care of the carry-overs, it is necessary to have another set of AND and OR devices except that the AND devices are wired differently. See Fig. B-7.

Numbers 3, 5, 6, and 7 would be wired to OR device 2, since they are the conditions which produce a carry requirement. Numbers 0, 1, 2, and 4 would be connected to OR device 1, since they do not produce a carry requirement. Whereas the outputs of the A set are connected to the register to record the sum, the outputs of the C set are connected to the C's of both sets, after being delayed one-cycle. This delay is so that the carry outputs of one Top and Bottom pair arrive at the AND devices at the same time that the following pair of numbers arrive.

In summary then, the two numbers, Top and Bottom, are successively read, position-by-position (one cycle each), into the AND and OR devices, and the A's appear one-by-one in successive positions of the register; that is, the first A appears in the rightmost position of the register; the next A would be shifted one place to the left; the next, one place to the left of that; and so on. (This shifting can be accomplished electronically.)

Fig. B-8 shows, in block diagram form, the connections between the A circuits and the C circuits. There are many details not included and there are different and better methods of doing the same job of adding binary numbers in actual practice. However, the purpose is to demonstrate that from the point of view of logical processes which can also be iterative, the concept of coincidence, like comparison, is important.

Calculation of Pseudo-Random Numbers

In Chapters 3 and 4 the use of random numbers was discussed, but the method of obtaining random numbers was not mentioned. There are several ways of generating random numbers, two of which are to roll a die or to flip a coin.

It is often desired to use random numbers when solving mathematical problems with the aid of a digital computer. One way in which this can be done is to enter into the computer a list of numbers previously selected by a random process. With such a list available in the storage unit of the computer, whenever the problem requires a random number the next one on the list is selected by the program.

There are advantages to having the computer generate its own random numbers as required, since it would then be unnecessary to use storage space to hold the list.

Several ways have been devised to generate such numbers. They are called *pseudo-random* numbers instead of random numbers since, as will be seen, the sequences are in a sense predictable. Random number sequences, on the other hand, by definition, are not predictable.

The method to be described can be seen to be repetitive and therefore nicely adaptable to a digital computer. The method can be ineffective under some circumstances when zeroes appear at certain places in the sequence. The method to be described is called the "center squaring" method. It consists of selecting a number, squaring it, selecting the center digits of the result, squaring this new number, etc.

For example, select the number 68342797, and square it.

$$\begin{array}{r} 68342797 \\ \times\ 68342797 \\ \hline 4670737901783209 \end{array}$$

Now select the eight digits in the center of this product and square the number thus formed:

$$\begin{array}{r} 73790178 \\ \times\ 73790178 \\ \hline 5444990369271684, \end{array}$$

and, once more:

$$\begin{array}{r} 99036927 \\ \times\ 99036927 \\ \hline 9808312909603329. \end{array}$$

After the three cycles, the following numbers would be generated.

68342797
73790178
99036927
31290960

If, as in Example 5 of Chapter 4, random numbers were required they could be selected by this method using two-digit pairs as follows: 68,34,27,97,73,79,01,78, etc. Or they could be selected in another order such as 97,78,27,60,27,01,69,09, etc.

As stated before, any particular generation may not be a good one and may result, for example, in large numbers of zeros. If too many zeros show up in a given product, subsequent squaring processes may cause the entire procedure to start producing nothing but zeros. To guard against this possibility that a poor sequence might be generated, the computer can be programmed to check the sequence of pseudo-random numbers it produces.

One kind of checking procedure is simply to count the separate digits of the entire sequence and determine if the distribution of them is what would be expected. For example, if 1000 digits were examined and it was found that there were about equal numbers of each of the digits, there would be more reason to believe that this sequence was a random one than if the distribution contained no 3's or 7's and four-hundred 6's.

There are other ways of checking sequences that are dependent upon internal cycles such as the frequency of certain pairs of numbers or short sequences of three or four digits which appear often. It is also possible to get another measure of the characteristics of a sequence by determining if there are about the same number of odd numbers as even numbers.

These tests are developed from the fields of statistics and probability and are mentioned only to suggest general procedures. It is possible to program the computer to perform such tests on the numbers it generates and according to criteria specified by the writer of the program exclude sequences that are not good ones.

Some Classroom Techniques

There are various devices available to teachers and students which may be useful in learning the fundamentals of computers. The reader is urged to consider whether such devices are characteristic of stored-program computers. If they are not stored-program devices, their limitations should be recognized. Among the devices available are the following:

PAPAC, a do-it-yourself paper computer. This device is described in *Communications of the Association for Computing Machinery*, Vol. 2, No. 9, September 1959. It consists of a simple two-register, one-bit, fixed instruction binary digital computer which can be built in less than one hour from plans shown in the article, using only 3 dozen common pins, a tube of glue, and a pair of scissors to cut out the plans.

MINIVAC, a small electro-mechanical (relay) type digital computer which includes many of the basic ideas incorporated in large-scale information-processing machines. Input and output may be either decimal or binary. It is available from Scientific Development Corp., 372 Main Street, Watertown, Mass.

BRAINIAC, described as "the smallest and lowest-cost semi-automatic, general-purpose digital computer existing," comes in a kit consisting ot wire, switches, bulbs, sockets, etc., which are assembled by the user. It is available from Berkeley Enterprises, Inc., 815 Washington Street, R-224, Newtonville 60, Mass.
Other do-it-yourself computers are described in references 36 and A10 in the Bibliography.

Classroom Simulation of a Digital Computer and Flow Diagram

A teaching technique for use in the introduction of the ideas of flow diagramming is based on one which was used to teach the internal operation of the IBM 650 computer to a group of ninth grade algebra students. The technique consists of the simulation of various parts of the machine by student groups who then follow the steps in a previously written program to calculate the sum of the first one-hundred integers. Each group is given the responsibility for performing the operations of some part of the computer such as the arithmetic register, the accumulator, the program register, etc. The groups are supplied with instruction sheets that detail the operations they are to perform, and they record the results of their operations on slips of paper which are passed from one group to another to simulate information transfer. The control group keeps track of the status of each other group and does not permit one operation to start until a preceding one is completed. The students show considerable enthusiasm for the "game," and are eager to write programs of their own for the "computer" after they understand what is happening. (See reference 10 in the Bibliography.)

The same idea is applicable to the teaching of flow diagramming and is useful in demonstrating the dynamic nature of processes for which flow diagrams are useful.

To implement the idea, each box in an existing flow diagram might be assigned to a group of students with instructions to perform the operation indicated in the box when a slip of paper arrives at their group from some other group. This procedure is probably of greatest value for younger students since to them the game aspect of the process is most appealing.

Classroom Simulation of a Serial Binary Counter

The most important element of a serial binary counter is the flip-flop. This is a device which when impulsed changes its state, whatever it may be. That is, if it is "off" when impulsed it is then turned "on". It it was "on" it is turned "off" when impulsed. "On" might mean to turn a light on, and "off" might mean to turn the light off. Its other characteristic of interest here is that when it is "on," and then impulsed, it *will* pass the impulse to another flip-flop element. When it is "off" and impulsed it *will not* pass the impulse.

The flip-flop follows these rules then:

1. Elements change state when impulsed.
2. Elements pass impulses they receive if they were "on" at the time.

Three flip-flop stages are represented below by three circles.

All stages are "off" initially. The first impulse will change stage No. 1 from "off" to "on" but will not pass the impulse. Therefore, after impulse No. 1, the three stages are

Off Off On.

The next impulse will turn No. 1 "off" but this time will pass the impulse to No. 2 and it will change to "on". They now look like this:

Off On Off.

The next impulse will turn No. 1 "on" but not pass the impulse. The three stages are now:

Off On On.

The next impulse will turn No. 1 "off" and pass the impulse which will turn No. 2 "off." No. 2 will also pass the impulse to No. 3 and turn it "on." They are now as follows:

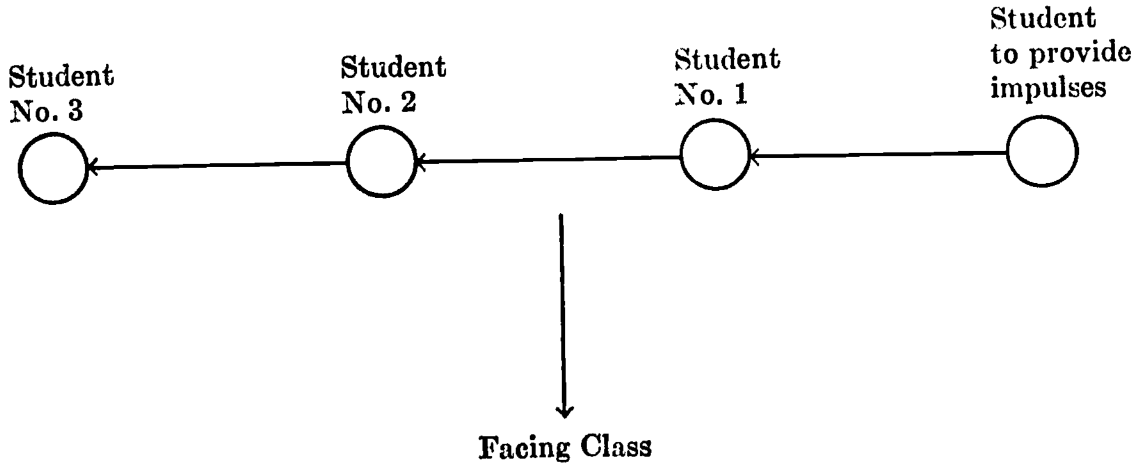
On Off Off.

If, in the above, all the "on's" are replaced by 1's, and all of the "off's" replaced by 0's, it will be seen that the binary numerals corresponding to 1, 2, 3, 4 have been developed.

This procedure can be simulated in the classroom by having one student represent each of the stages. The students should stand about 2 feet apart, in a straight line, facing the class. The state "off" will be represented by "arms down to the sides." The state "on" will be represented by "right arm up."

The impulse will be a touch on the left shoulder. The source of impulses can be another student. The passing of impulses will be done by a student touching the left shoulder of the student at his right (the next stage) whenever he changes from "on" to "off" (lowers his upraised arm). This student simulator of a binary counter will produce the 1's and 0's (arms up or arms down) which correspond to the binary numerals.

The diagram below indicates the student arrangement for the simulation of the counter.



Appendix C

BIBLIOGRAPHY

Books and Pamphlets

1. ADLER, I. *Thinking Machines.* New York: The John Day Co., 1961. (Signet Science Library, paperback edition.)
2. ALLENDOERFER, C. B., and OAKLEY, C. O. *Principles of Mathematics.* New York: McGraw-Hill, 1955.
3. ALT, FRANZ L., and OTHERS. *Advances in Computers.* New York: Academic Press, 1960.
4. ALT, FRANZ L. *Electronic Digital Computers, Their Use in Science and Engineering.* New York: Academic Press, 1958.
5. ANDRÉE, RICHARD V. *Programming the IBM 650 Magnetic Drum Computer and Data Processing Machine.* New York: Henry Holt and Co., 1958. Paperback.
   Introductory course in programming the IBM 650.
6. ARONSON, M. H. *Computer Handbook.* Pittsburgh: The Instruments Publishing Co., 1955.
   Describes basic functions, operations and applications of analog and digital computers.
7. BARTEE, THOMAS C. *Digital Computer Fundamentals.* New York: McGraw-Hill Book Co., 1960.
8. BELLMAN, R. *Adaptive Control Processes—A Guided Tour.* (Rand Corporation) Princeton, N. J.: Princeton University Press, 1961.
9. BERKELEY, E. C. *Giant Brains or Machines That Think.* New York: John Wiley and Sons, 1949.
10. BERKELEY, E. C., and WAINWRIGHT, L. *Computers—Their Operation and Applications.* New York: Reinhold Publishing Corp., 1956.
    Describes small and large computers. Contains a glossary and a bibliography.
11. BOOLE, GEORGE. *An Investigation of the Laws of Thought on Which Are Founded the Mathematical Theories of Logic and Probabilities.* New York: Dover Publications, 1953.
    A reprint of the 1854 classic—*Boolean Algebra.*
12. BOOTH, A. D.; BRANDWOOD, L.; and CLEAVE, J. P. *Mechanical Resolution of Linguistic Problems.* New York: Academic Press, 1958.
13. CHAPIN, NED. *An Introduction to Automatic Computers.* Princeton, N. J.: D. Van Nostrand Co., 1957.
14. COMMISSION ON MATHEMATICS. Publications available from College Entrance Examination Board. Educational Testing Service, Box 592, Princeton, N. J.

(a) *Report of the Commission on Mathematics.* (Out of print)
(b) *Appendices.* (Out of print)
(c) *Introductory Probability and Statistical Inference.* (Out of print)
(d) *Concepts of Equation and Inequality.* (Out of print)
(e) *Informal Deduction in Algebra.* (Out of print)

15. Courant, R., and Robbins, H. *What is Mathematics?* London: Oxford University Press, 1941.
16. Crowder, Norman A. *The Arithmetic of Computers.* Garden City, N. Y.: Doubleday and Co., 1960.
    Designed to teach the binary and octal number systems through self study. A programmed textbook.
17. Culbertson, J. T. *Mathematics and Logic for Digital Devices.* New York: D. Van Nostrand, 1958.
    Textbook for undergraduate students who intend to enter the computer field.
18. Dantzig, Tobias. *Number, the Language of Science.* 4th edition. Garden City, N. Y.: Doubleday and Co., 1956. Paperback edition.
19. Davison, J. S. *Programming for Digital Computers.* New York: Gordon and Breach, 1961.
20. Diebold, John. *Automation, the Advent of the Automatic Factory.* Princeton, N. J.: D. Van Nostrand Co., 1952.
21. Dwyer, Paul S. *Linear Computations.* New York: John Wiley and Sons, 1951.
22. Eckert, W. J., and Jones, R. *Faster, Faster: A Simple Description of a Giant Electronic Calculator and the Problems It Solves.* New York: McGraw-Hill Book Co., 1956.
    The computer described is the IBM NORC (Naval Ordnance Research Calculator).
23. Ferguson, R. O., and Sargent, L. F. *Linear Programming: Fundamentals and Applications.* New York: McGraw-Hill Book Co., 1958.
    Major methods of solution and important applications of linear programming are explained.
24. Galler, Bernard A. *The Language of Computers.* New York: McGraw-Hill Book Co., 1962.
    Introduction to *languages* by means of which communication with computers is possible. The language described is MAD (Michigan Algorithm Decoder).
25. Gotlieb, C. C., and Hume, J. N. P. *High Speed Data Processing.* New York: McGraw-Hill Book Co., 1958.
26. Gorn, S.; Manheimer, W.; and Brandwein, P. *The Electronic Brain and What It Can Do.* Chicago: Science Research Associates, 1956. (Out of print)
27. Greenberger, M., and others. *Management and the Computer of the Future.* New York: John Wiley and Sons, 1962.
28. Gregory, R. H., and Van Horn, R. L. *Automatic Data Processing: Principles and Procedures.* San Francisco: Wadsworth Publishing Co., 1960. Text edition.

Thorough coverage of concepts in data processing, computing equipment, programming, and systems design.

29. HALSTEAD, MAURICE H. *Machine Independent Computer Programming* Washington, D.C.: McGregor and Werner, 1962.
   This book is suitable for beginners and experienced persons. It describes the NELIAC (Naval Electronics Laboratory International Algol Compiler) language. A section on compiling compilers and compiling systems is included.
30. HAMMER, PRESTON C., editor. *The Computing Laboratory in the University.* Madison: University of Wisconsin Press, 1957.
   High school teachers may find this a very helpful book for gaining insight into the computing industry's equipment and problems, and the industry's effects on school curricula, personnel needs, etc.
31. HARTREE, D. R. *Calculating Instruments and Machines.* Urbana, Ill.: University of Illinois Press, 1949.
32. HASTINGS, CECIL, JR. *Approximations for Digital Computers.* Princeton, N. J. Princeton University Press, 1955.
   An excellent reference book of approximation formulas.
33. HEIN, LEONARD W. *Introduction to Electronic Data Processing for Business.* Princeton, N. J.: D. Van Nostrand Co., 1961.
34. HILDEBRAND, F. B. *Introduction to Numerical Analysis.* New York: McGraw-Hill Book Co., 1956.
   Many sample problems are included in the textbook.
35. INTERNATIONAL BUSINESS MACHINES CORPORATION. Publications available from local IBM branch offices.
   (a) *1401 Data Processing System: General Information Manual*, D24-1401-2.
   (b) *An Introduction to Information Retrieval*, E20-8044. 1960.
   (c) *The COBOL Translater*, F28-8053. 1960.
      A number of companies are implementing their data processing equipment with the COBOL language.
   (d) *Data Processing Bibliography*, J20-8014-2. 2nd revision, 1961.
   (e) *Flow Charting and Blocking Diagramming Techniques*, C20-8008. 1959.
   (f) *FORTRAN: General Information Manual*, F28-8074.
   (g) *General Information Manual: An Introduction to Engineering Analysis for Computers*, F20-8077. 1961.
   (h) *Introduction to IBM Data Processing Systems*, F22-6517. 1960.
      Used in introductory IBM educational programs to provide a basic understanding of computer systems.
   (i) *Principles of Programming*, F320-1461. 1961.
   (j) *World of Numbers*, F520-0309. 1958.
      Excellent reading for young students: history of numbers and their importance to man.
   (k) *Data Processing Courses in Vocational and Secondary Schools*, F20-8087. 1962.
36. IRWIN, WAYNE C. *Digital Computer Principles.* Princeton, N. J.: D. Van Nostrand Co., 1960.

37. KORN, G. A., and KORN, T. M. *Electric Analog Computers*, 2nd edition. New York: McGraw-Hill Book Co., 1956.

38. LANGER, SUSANNE K. *An Introduction to Symbolic Logic*, 2nd edition. New York: Dover Publications, 1953. Paperback.
   Intended for the layman whose background in mathematics and logic is elementary.

39. LEEDS, H. D., and WEINBERG, G. M. *Computer Programming Fundamentals*. New York: McGraw-Hill Book Co., 1961.
   Emphasizes basic concepts, ideas, and techniques required for communication with computers.

40. LOCKE, W. N., and BOOTH, A. D. *Machine Translation of Languages*. New York: John Wiley and Sons, 1955.

41. MACE, DAVID, and ALSOP, JOYCE. *A Simplified System for the Use of an Automatic Calculator*. New York: Watson Scientific Computing Laboratory, 1957. (Out of print)
   Useful to the young student who wants to learn something about programming.

42. MACMILLAN, R. H. *Automation: Friend or Foe?* New York: Cambridge University Press, 1956.
   Describes the use of computers for the benefit of the layman.

43. MATHEMATICAL ASSOCIATION OF AMERICA. *Professional Opportunities in Mathematics*. Buffalo, N. Y.: University of Buffalo, 1961.

44. McCORMICK, E. M. *Digital Computer Primer*. New York: McGraw-Hill Book Co., 1959.
   A student does not require extensive training in mathematics in order to read this textbook. Excellent introduction to the operation and organization of a digital computer.

45. McCRACKEN, D. D. *Digital Computer Programming*, text edition. New York: John Wiley and Sons, 1957.
   Introductory textbook for digital computer programming.

46. McCRACKEN, D. D.; WEISS, H.; and LEE, TSAI-HWA. *Programming Business Computers*, text edition. New York, John Wiley and Sons, 1959.
   Includes numerous examples of basic concepts for programming business computers.

47. MILNE, W. E. *Numerical Calculus*. Princeton, N. J.: Princeton University Press, 1949.
   Approximations, interpolations, finite differences, numerical integration, curve fitting.

48. MURPHY, J. S. *Basics of Digital Computers* (3 vols.). New York: John F. Rider Publishers, 1958.
   A well-illustrated introduction to computers.

49. NATHAN, R., and HANES, E. *Computer Programming Handbook*. Englewood Cliffs, N. J.: Prentice-Hall, 1962.
   A guide for beginners.

50. NATIONAL EDUCATION ASSOCIATION (departmental publications). 1201 Sixteenth St., N.W., Washington, D.C. 20036

(a) National Council of Teachers of Mathematics. *Insights Into Modern Mathematics*. 23rd Yearbook. Washington, D.C.: the Council 1957.
(b) National Council of Teachers of Mathematics. *Careers in Mathematics*. Washington, D.C.: the Council, 1961.
(c) National Science Teachers Association. *Encouraging Future Scientists: A Selected Bibliography*. Washington, D.C.: the Association, 1959.

51. NEWMAN, JAMES R. *The World of Mathematics* (4 vols.). New York: Simon and Schuster (paperback edition), 1960.
Excellent material for high school students.
52. NIELSEN, KAJ L. *Methods in Numerical Analysis*. New York: Macmillan Co., 1956.
College elementary textbook on the su ᐩ of numerical analysis.
53. PETERS, JEAN. *Seven Place Values of 'ɪ rigonometric Functions for Every Thousandth of a Degree*. New York: D. Van Nostrand Co., 1942.
54. PHISTER, M., JR. *Logical Design of Digital Computers* (text edition). New York: John Wiley and Sons, 1958.
Carefully written textbook on digital computers for engineers and designers.
55. POSTLEY, J. A. *Computers and People*. New York: McGraw-Hill Book Co., 1960.
56. RALSTON, A., and WILF, H. S. *Common Mathematical Methods for Digital Computers*. New York: John Wiley and Sons, 1960.
57. REINFELD, N. V., and VOGEL, W. R. *Mathematical Programming*, text edition. Englewood Cliffs, N. J.: Prentice-Hall, 1958.
An introductory textbook of linear programming suitable for a beginner or layman.
58. RICHARDS, R. K. *Digital Computer Components and Circuits*. New York: D. Van Nostrand Co., 1957.
Mostly of interest to engineers. Clear explanation of the design and workings of a computer.
59. RICHARDS, R. K. *Arithmetic Operations in Digital Computers*. New York: D. Van Nostrand Co., 1955.
60. SCARBOROUGH, J. B. *Numerical Mathematical Analysis*, 4th edition. Baltimore, Md.: Johns Hopkins Press, 1958.
Principles, methods, and processes used to obtain numerical ılts are clearly explained.
61. SOULE, GEORGE. *What Automation Does to Human Beings*. London: Sidgewick and Jackson, 1956. (Out of print.)
Optimistic examination of the social and economical consequences of automation.
62. SIEGEL, PAUL. *Understanding Digital Computers*. New York: John Wiley and Sons, 1961.
63. SPERRY RAND CORPORATION. Blue Bell, Pa. Univac Educational Series, 1957–59.
(a) *How the Computing System Works for You, No. 1.*
(b) *Programming—A New Profession for You, No. 2.*

(c) *An Annotated Bibliography, No. 3.*
Excellent materials for high school students.

64. STANTON, RALPH G. *Numerical Methods for Science and Engineering.* Englewood Cliffs, N. J.: Prentice-Hall, 1960.
65. STIBITZ, G. R., and LARRIVEE, J. A. *Mathematics and Computers.* New York: McGraw-Hill Book Co., 1957. Paperback and hard-cover editions.
The book is written in a manner that will interest students and laymen.
66. STIFLER, W. W., editor. *High Speed Computing Devices.* New York: McGraw-Hill Book Co., 1950.
This textbook is mostly of historical interest.
67. SYSTEMS DEVELOPMENT CORPORATION. Publications office: 2500 Colorado Ave., Santa Monica, Calif.
    (a) Vaughn, L., and Pelton, W. *Programming: An Approach to Problem Solving,* Report SP-167.
    (b) Dobbs, G.; Myer, E.; and Perstein, M. *Computer Programming and Advanced Mathematical Concepts for Elementary School Students,* Report TM-494.
68. TAUBE, MORTIMER. *Computers and Common Sense.* New York: Columbia University Press, 1961.
69. THRALL, R. M.; COOMBS, C. H.; and DAVIS, R. L. *Decision Processes.* New York: John Wiley and Sons, 1954.
70. U. S. GOVERNMENT. Publications available from Government Printing Office, Washington 25, D. C.
    (a) WEIK, M. H. *A Third Survey of Domestic Electronic Digital Computing Systems.* Office of Technical Services, Department of Commerce, March 1961.
    (b) *Occupations in Electronic Data Processing Systems.* Prepared by Occupational Analysis Branch, U. S. Employment Service, Department of Labor, 1959.
    (c) *Report on the Use of Electronic Data Processing Systems in the Federal Government.* U. S. Congress, House of Representatives, Committee on Post Office and Civil Service, 1960.
    (d) *Mathematics and Your Career.* Bulletin No. 0-542720. Department of Labor, 1960.
    (e) *Employment Outlook for Mathematicians, Statisticians, Programmers.* Bulletin No. 1255-46.
71. USPENSKY, JAMES V., and HEASLET, N. A. *Elementary Number Theory.* New York: McGraw-Hill Book Co., 1939.
72. VON NEUMANN, JOHN. *The Computer and the Brain.* New Haven, Conn.: Yale University Press, 1958.
Notes from the Silliman Memorial Lecture given by an outstanding mathematician. The text is of general interest.
73. WASS, C. A. A. *Introduction to Electronic Analog Computers.* New York: McGraw-Hill Book Co., 1955.
74. WRUBEL, MARSHALL, H. *A Primer of Programming for Digital Computers.* New York: McGraw-Hill Book Co., 1959.
An introduction to computer programming based on the IBM 650.

(c) *An Annotated Bibliography, No. 3.*
Excellent materials for high school students.

64. STANTON, RALPH G. *Numerical Methods for Science and Engineering.* Englewood Cliffs, N. J.: Prentice-Hall, 1960.
65. STIBITZ, G. R., and LARRIVEE, J. A. *Mathematics and Computers.* New York: McGraw-Hill Book Co., 1957. Paperback and hard-cover editions.
The book is written in a manner that will interest students and laymen.
66. STIFLER, W. W., editor. *High Speed Computing Devices.* New York: McGraw-Hill Book Co., 1950.
This textbook is mostly of historical interest.
67. SYSTEMS DEVELOPMENT CORPORATION. Publications office: 2500 Colorado Ave., Santa Monica, Calif.
(a) Vaughn, L., and Pelton, W. *Programming: An Approach to Problem Solving,* Report SP-167.
(b) Dobbs, G.; Myer, E.; and Perstein, M. *Computer Programming and Advanced Mathematical Concepts for Elementary School Students,* Report TM-494.
68. TAUBE, MORTIMER. *Computers and Common Sense.* New York: Columbia University Press, 1961.
69. THRALL, R. M.; COOMBS, C. H.; and DAVIS, R. L. *Decision Processes.* New York: John Wiley and Sons, 1954.
70. U. S. GOVERNMENT. Publications available from Government Printing Office, Washington 25, D. C.
(a) WEIK, M. H. *A Third Survey of Domestic Electronic Digital Computing Systems.* Office of Technical Services, Department of Commerce, March 1961.
(b) *Occupations in Electronic Data Processing Systems.* Prepared by Occupational Analysis Branch, U. S. Employment Service, Department of Labor, 1959.
(c) *Report on the Use of Electronic Data Processing Systems in the Federal Government.* U. S. Congress, House of Representatives, Committee on Post Office and Civil Service, 1960.
(d) *Mathematics and Your Career.* Bulletin No. 0-542720. Department of Labor, 1960.
(e) *Employment Outlook for Mathematicians, Statisticians, Programmers.* Bulletin No. 1255-46.
71. USPENSKY, JAMES V., and HEASLET, N. A. *Elementary Number Theory.* New York: McGraw-Hill Book Co., 1939.
72. VON NEUMANN, JOHN. *The Computer and the Brain.* New Haven, Conn.: Yale University Press, 1958.
Notes from the Silliman Memorial Lecture given by an outstanding mathematician. The text is of general interest.
73. WASS, C. A. A. *Introduction to Electronic Analog Computers.* New York: McGraw-Hill Book Co., 1955.
74. WRUBEL, MARSHALL, H. *A Primer of Programming for Digital Computers.* New York: McGraw-Hill Book Co., 1959.
An introduction to computer programming based on the IBM 650.

75. YOUNG, FREDERICK H. *Digital Computers and Related Mathematics*. New York: Ginn and Co., 1961.
Good material for high school students.
76. ANONYMOUS. *Data Processing by Electronics*. New York: Haskins and Sels Publishers, 1955.
Includes binary arithmetic, theory of operation, and the operating characteristics of various computers.

Articles

1. BUCHMAN, AARON L. "A High School Digital Computer Program." *Mathematics Teacher*, March 1958. pp. 176–80.
2. DODES, I. A. "Numerical Analysis as a Twelfth-Year Elective." *Mathematics Teacher*, May 1962 pp. 334–36.
3. ENGLEBART, DOUGLAS C. "Games That Teach the Fundamentals of Computer Operation." *IRE Transactions*. Vol. E.C.-10, No. 1, March 1960.
Reprints may be obtained from WGBE, (the Working Group for Better Education) 815 Washington St., Newtonville 60, Mass.
4. DURKEE, R. M. "How Much . . . or . . . How Many?" *Mathematics Teacher*, March 1958. pp. 169–75.
5. GOLDSTINE, H. H. "Calculating Machines." *Encyclopaedia Britannica*, Vol. 4, 1961.
6. GROSSMAN, GEORGE. "A Report of a National Science Foundation Summer Institute in Mathematics for High School Students at Columbia." *Mathematics Teacher*, February 1961. pp. 75–81.
7. GROSSMAN, GEORGE. "Probability and Statistics in the Twelfth Year?" *Mathematics Teacher*, November 1961. pp. 540–46.
8. HART, DONALD S. "What is a Digital Computer?" *General Motors Engineering Journal*. Vol. 5, No. 2, April-May-June, 1958.
9. HELLER, GEORGE C. "A Computer Curriculum for the High School." *Datamation*, May 1962, p. 23.
10. HORTON, GEORGE W. "HOBLADIC." *Mathematics Teacher*, April 1961. pp. 212–16.
11. HOSHEL, MARGARET. "A Survey: The Advent of the Automatic Computer and Its Effect on Secondary Education." Unpublished Report. Ann Arbor: University of Michigan, May 1962.
12. KING, GILBERT, W. "Information." *Scientific American*, September 1952. pp. 132–42.
13. LITTLEFIELD, D. G. "Computer Programming for High Schools and Junior Colleges." *Mathematics Teacher*, April 1961. pp. 220–23.
14. MANHEIMER, WALLACE. "A Club Project in a Modern Use of Computers." *Mathematics Teacher*, May 1957. pp. 350–55.
15. REES, MINA. "The Impact of the Computer." *Mathematics Teacher*, March 1958. pp. 162–68.
16. TILLITT, HARLEY E. "Computer Programming for Young Students." *Journal of the Association for Computing Machinery*. Vol. 5, No. 4, October 1958. pp. 309–18.

17. Van Tassel, L. T. "Digital Computer Programming in High School Classes." *Mathematics Teacher*, April 1961. pp. 217–19.
18. Wilcox, Marie S. "Mathematics and Industry . . . A Report." *Mathematics Teacher*, March 1960. pp. 207–08.
19. Wood, Thomas. "How an Analog Computer Works." *SAE* (Society of Automotive Engineers) *Journal*. Vol. 63, August 1955.

Professional Organizations

1. American Institute of Electrical Engineers
   29 West 29th Street
   New York 18, New York
2. Armour Research Foundation of Illinois
   Institute of Technology
   CAS: 2, Technology Center
   Chicago 16, Illinois
3. Association for Computing Machinery
   2 East 63rd Street
   New York 21, New York
4. The Institute of Radio Engineers
   Professional Group on Electronic Computers
   1 East 79th Street
   New York 21, New York
5. National Academy of Sciences
   National Research Council
   Constitution Avenue
   Washington 25, D. C.
6. National Council of Teachers of Mathematics
   1201 Sixteenth Street, N.W.
   Washington, D.C. 20036
7. Society for Industrial and Applied Mathematics
   Box 7541
   Philadelphia 1, Pennsylvania

Periodicals

1. *Automation*, monthly. Penton Publishing Co., Cleveland 13, Ohio.
2. *Communications of the ACM*, monthly. Association for Computing Machinery, 2 East 63rd St., New York 21, N. Y.
   Journal of the ACM, quarterly. Association for Computing Machinery, 2 East 63rd St., New York 21, N. Y.
3. *Management and Business Automation*, monthly. Office Appliance Co., 600 W. Jackson Blvd., Chicago 6, Ill.
4. *Computers and Automation*, monthly. Berkeley Enterprises, Inc., 815 Washington St., Newtonville 60, Mass.
5. *Datamation*, bimonthly. 141 East 44th St., New York 17, N. Y.
6. *The Mathematics Teacher*, official journal of the National Council of Teachers of Mathematics, 1201 Sixteenth St., N.W., Washington, D.C. 20036. Pub-

lished eight times during school year, $5.00 per year includes membership in Council; $1.50 to students.

7. *Scientific American*, monthly. 415 Madison Ave., New York 17, N. Y.
8. *Journal of the Society for Industrial and Applied Mathematics*, quarterly. Box 7541, Philadelphia 1, Pa.
9. *SIAM Review*, semiannually. Society for Industrial and Applied Mathematics, Box 7541, Philadelphia 1, Pa.
10. *SIAM Newsletter*, monthly. Society for Industrial and Applied Mathematics, Box 7541, Philadelphia 1, Pa.

Appendix D

GLOSSARY

absolute value. The absolute value of a number is a number which is calculated by the following procedure: For each real number, x, the absolute value of x, written $|x|$, is equal to x if x is greater than or equal to zero, and is equal to $-x$ if x is less than zero. Symbolically:

$$|x| = x \text{ if } x \geq 0.$$
$$|x| = -x \text{ if } x < 0.$$

accumulator. A device containing a storage register where results are accumulated.

address. A label, name, or number which designates a register, a location, or a device where information is stored; the part of an instruction which specifies the location of an operand.

ALGOL. An international language for use in writing algorithms. The name **ALGOL** is from *ALGO*rithmic *L*anguage. See *compiler*.

algorithm. A sequence of well-defined steps leading to the solution of a problem.

analog computer. A computer which represents variables by physical analogies in continuous form, such as amount of rotation of a shaft, amount of voltage, etc. Contrasted to digital computer: the difference is sometimes expressed by saying that an analog computer *measures* whereas a digital computer *counts*.

AND circuit. An electronic circuit which provides an output signal only if all of the input signals for which it is designed are available simultaneously.

arithmetic operation. A computer operation in which numerical calculations form the basis of the procedure, the result of which has a number as its value.

arithmetic unit. That component of a computer where arithmetic and logical operations are performed.

automatic coding. A technique whereby a machine translates a routine written in symbolic coding to one consisting of machine language.

binary coded decimal. A form of notation by which each decimal digit of a numeral is converted into a pattern of 1's and 0's. For example, the decimal numeral for the number 12 is coded as: $\underset{(1)}{0001}\ \underset{(2)}{0010}$. This is also used for alphabetic symbols and for punctuation symbols.

binary digit. A digit in the binary scale of notation; this digit may be only 0 (zero) or 1 (one). Sometimes called a "bit."

binary number. A number whose name is written in binary notation.

bit. A binary digit.

block diagram. See *flow diagram*.

Boolean algebra. An algebra dealing with classes, propositions, on-off elements, etc., associated by operators AND, OR, NOT, EXCEPT, IF, THEN, etc.

(See *logical operations.*) Named after George Boole, famous English mathematician (1815–64).

branch. See *conditional transfer of control.*

buffer storage. Any device which temporarily stores information during a transfer of information. From a programming standpoint, it refers to a device for matching the speeds of internal computation and an input or output device, thereby permitting simultaneous computation and input or output.

card. Pasteboard adapted for being punched in a pattern to which meaning may be assigned. The punched holes are sensed mechanically by metal fingers, electrically by wire brushes, or photoelectrically, depending upon the particular device.

card punch. A machine containing a keyboard something like that of a typewriter and used to punch holes in cards. Also called "keypunch."

card reader. A machine, operating mechanically, electrically, or photoelectrically to sense the presence of holes in cards and to translate the hole pattern to (usually) a set of electrical signals to be used by a computer, or other device.

cell. A storage location.

character. One of a set of elementary symbols which may be arranged in ordered groups to express information; these symbols may include the decimal digits 0 through 9, the letters A through Z, punctuation symbols, special input and output symbols, and any other symbols which a computer may accept.

COBOL (COmmon Business Oriented Language). A language for writing programs to solve business problems, and which, to some extent, is common to more than one kind of computer. See *compiler.*

code (noun). The system of symbols used in preparing instructions for the computer.

code (verb). To write instructions for a computer, either in machine language or some other language.

collate. To merge items from two or more similarly sequenced files into one sequenced file without necessarily including all items from the original files.

compile. To produce a machine language program from a program written in some compiler language.

compiler. A special program for a given machine which will translate instructions written in a particular program language into machine language instructions for that machine.

compiler language. A well-defined language closely related to the language in which a problem is originally stated (e.g., English or mathematics). Used to write programs which are then automatically translated by a compiler into a machine language program.

computer. Any device capable of accepting information, processing it, and providing the results of the processing in acceptable form. In this text the term is most often meant to imply a *stored program digital computer.*

conditional transfer of control. A point in a routine where one of two or more choices can be taken as the result of a previous occurrence in the routine.

core storage. A form of high-speed storage in which binary information is represented by the direction of magnetization of ferromagnetic cores.

cybernetics. The comparative study of the control and internal communication systems of information-handling machines and the central nervous systems of animals and men, in order to understand better the functioning of information transfer and processing in such systems.

data processing. A generic term for all of the operations carried out on data according to precise rules of procedure; a generic term for computing in general as often applied to business situations.

digital computer. A computer in which information is represented in discrete form, such as by one of two directions of magnetization of a magnetic core, or by the presence or absence of an electric pulse at a certain point in time. See *analog computer*.

error. The amount of loss of precision in a quantity; the difference between an accurate quantity and its calculated (measured) approximation. *Errors* occur in numerical methods; *mistakes* occur in programming, coding, data transcription, and operation of computers; *malfunctions* occur in computers because of failures in the performance of machine components.

field. A set of one or more characters treated as a whole; a unit of information. A collection of characters.

file. A collection of records; an organized collection of information directed toward some purpose.

fixed point calculation. A system of handling numbers which uses a fixed or constant location of the point which separates the whole numbers from fractions. Applies to binary, decimal, or other systems.

flip-flop. An electronic device having but two stable states and thus capable of storing one binary digit for each state, e.g., "1" or "0".

floating point calculation. A system of handling numbers which for a given number uses a separate number to specify the point which separates the whole part from the fractional part. A decimal number, for example $-638,020,000.$, might be represented as -6.3802×10^8.

flow diagram. A graphic representation of a sequence of operations required to carry out some procedure.

hardware. The mechanical, magnetic, electric and electronic devices from which a computer is constructed; or the computer itself as contrasted with its programs and programming systems. See *software*.

initialize. To execute the instructions immediately prior to a loop, which set addresses, counters, etc., to their desired initial values. See also *loop*.

input. Information transferred from auxiliary or external storage into the internal storage of a computer.

input unit. A device to transfer information from punched cards, magnetic tape, or other sources to the electrical impulses used in a computer.

instruction. A set of characters which as a unit causes the computer to perform one or more of its operations.

iterative procedure. The repetition of a sequence of steps with one or more variables being assigned different values for each step. These different values may be developed by the procedure itself. The repetition is stopped, by the procedure itself, on the basis of previously programmed numerical checks of its progress.

jump. See *unconditional transfer*.

key punch. See *card punch.*

library. An organized collection of standard and proven routines and subroutines which may be incorporated into larger routines.

logical diagram. See *flow diagram.*

logical operations. Those operations of a computer which are not arithmetic and not part of input or output, the results of which have *true* or *false* as their values.

loop. A coding technique whereby a group of instructions is repeated with modification of some of the instructions within the group and/or with modification of the data being operated upon. Usually consists of initialization, computing, modification, and testing, although not necessarily in that order. (Testing refers to the part of the procedure which determines if the process has been completed.)

machine language. A language for writing instructions in a form to be executed directly by the computer. Contrasted to symbolic coding languages or other non-machine languages. See also *compiler language.*

magnetic core. See *core storage.*

magnetic tape storage. A storage system in which information is recorded on the magnetizable surface of a strip of plastic tape.

malfunction. Failure in precision of performance of machine components. Also see *error.*

memory. See *storage.*

merge. To combine items from two or more similarly sequenced files into one sequenced file, including all items from the original files.

microsecond. One-millionth of a second.

millisecond. One-thousandth of a second.

mistake. See *error.*

mnemonic code. An operation code written in a symbolic notation that is easier to remember than the actual operation code of the machine. Must be converted to an actual operation code before execution. This conversion is done as part of another routine or program such as a compiler. (*MPY* could be such a code for the operation multiply.)

Monte-Carlo methods. Mathematical methods making use of the theory of probability and random numbers.

multi-precision (or multiple precision). A term used to indicate a program that will cause a computer to calculate using more digits in the calculations than the arithmetic units and individual storage locations were originally designed to use.

nanosecond. A billionth of a second.

'NOT' circuit. An electronic circuit which on a given cycle produces output if it did not receive input, and does not provide output when it does receive input. (Sometimes called an INVERTER.)

notation system for numbers. A systematic use of a particular set of numerals for the purpose of simplifying numerical calculations. An example is the familiar positional decimal notation using Arabic numerals. Other numeral symbols could be used to obtain a decimal positional notation.

numeration system. A set of symbols, such as Roman numerals or Arabic numerals, used to name numbers.

operation code. That part of an instruction designating the operation to be preformed.

OR circuit. An electronic circuit which provides output if any one of its inputs is active.

output. Information transferred from the internal storage of a computer to output devices or external storage.

overflow. The generation of a quantity beyond the capacity of a register; usually results in a machine stop, or conditional transfer, or an overflow indicator being turned on.

parameter. A variable in an expression, to which a value is assigned and which does not change while other variables in the expression take on values selected according to some particular sequence.

program (noun). A group of related routines that solve a given problem. (Sometimes meant to include flow diagrams as well as the computer instructions.)

program (verb). (1) To plan and organize a group of routines to solve a particular problem or set of related problems. (2) To write, in some language, the routines necessary for the solution of the problem. (Depending upon level of detail, this latter definition may sometimes be called coding.)

punched card. See *card*.

random numbers. A collection of numbers such that it is not possible to predict the next number by considering the previous numbers in the collection. Numbers selected by chance.

read. To transfer information from an input device to internal storage.

record (noun). A collection of fields.

register. (1) A device that can hold information while or until it is used. May consist of core storage. (2) A particular storage location usually in the arithmetic section of the computer.

rounding error. The error resulting from dropping certain less significant digits of a quantity and applying some adjustment to the more significant digits retained. Sometimes known as round-off error.

routine. A set of computer instructions that carries out some well-defined function.

simulator. A program or routine corresponding to a mathematical model or representing a physical model.

software. The programming systems required, in addition to the computer itself, such as a compiler.

sort (verb). To set records in sequence according to some key contained in the records; for example, in alphabetical or numerical sequence.

storage. Any device into which information can be transferred, which will hold information and from which the information can be obtained at a later time.

storage location. A place in storage where a unit of data or an instruction may be stored, identified by an address.

stored-program computer. A computer which can alter its own instructions in storage as though they were data and which can later execute the altered instructions.

subroutine. A routine which may be incorporated into a larger routine. Frequently part of the library of a computer installation.

symbolic coding. Coding in which instructions are written in non-machine language. That is, coding using symbolic notation for operators, operands, and locations instead of actual machine instruction codes and addresses.

symbolic logic. Reasoning involving non numerical relations often using special symbols. Also called "mathematical logic." *See Boolean algebra.*

systems analysis. The analysis of activity to determine precisely what must be accomplished and how to accomplish it.

transfer control. See *conditional transfer of control.*

translate. To change information from one form of representation, (language), to another without significantly affecting the meaning.

truncate. To cut off, as is done, for example, in a "truncated pyramid." In computing, to drop the digits beyond a certain point in the positional notation of a number without adjusting the remaining digits. See *truncation error* and *rounding error*.

truncation error. The error resulting from the use of only a finite number of terms of an infinite series.

unconditional transfer. An instruction to a computer to interrupt an established sequence and transfer control to some instruction out of the sequence.

word (machine). A set of characters having one addressable location and treated as one unit.

write. To transfer information from internal storage to an output device or to auxiliary storage.

Appendix E

A SHORT HISTORY OF COMPUTERS

The origins of mechanical aids to calculation are lost in the dim memories of the human race. However, the modern word "calculate," which is derived from a Latin word whose meaning was "stone" or "pebble," gives some hint as to a very early aid. It is probable that animals in a flock were counted by pairing one pebble with one animal, and thus a primitive mechanical aid to computation was instituted. The fact that Nature provided man with a computational aid (used in most classrooms even today) in the form of fingers should not be overlooked. It is no accident that the same word, "digit," is used to discuss parts of numerals as well as parts of hands.

Another well-known computing aid, the abacus (and its relatives), is of ancient origin and is still in use throughout the world. It may be the most efficient computation device, in terms of investment in equipment and ease of manufacture compared to the extent of computing aid obtained, that man has ever invented.

The history of modern mechanical aids to computation is generally considered to have started in about the middle of the 17th century when Blaise Pascal, a 19-year-old French mathematical genius, invented a device composed of gears and wheels which was capable of addition. The stylus-operated pocket adding machines now available are direct descendents of Pascal's calculating machine. Toward the end of the 17th century, Gottfried Wilhelm Leibniz, one of the two independent inventors of the modern calculus, proposed a machine which would multiply by repeated addition. Few further significant developments took place for the next hundred years.

During the early years of the 19th century, Joseph M. Jacquard developed a punched-card system to control complex weaving patterns on a loom of his invention. Shortly thereafter, using Jacquard cards to control sequences of operations, Charles Babbage in England began work on the most famous ancestor of the modern large-scale general-purpose computer, the difference engine.

Charles Babbage was a man whose ideas were about 100 years ahead of the technology of his day. The design of his difference engine contained ideas which include most of the basic elements of the modern general-purpose computer. Even though he was able to persuade the British government to advance him about £17,000 (almost $1,000,000 by today's standards) over a period of 10 years, and although he worked using his own money until his death, he was unable to produce a working machine. It is an unfortunate fact that, since the technology necessary for the realization of the difference engine did not develop for almost 100 years, most of Babbage's work was forgotten, and his ideas concerning applications of and programming for large-scale computers had to be rediscovered independently by the modern inventors of digital computers. Much of what we do know about Babbage and his work is due to the interest and efforts of Lady Ada Augusta, Countess of Lovelace, whose writings included also the description of what is now called "programming" in the modern sense of developing a set of instructions for a large-scale computer.

During the century following Babbage's first efforts little was accomplished in the area of large-scale computation. However, the adding machine and its relatives, desk calculators and accounting machines, as well as machines for handling and sorting data on punched cards were developed to a high order of effectiveness during this period. In fact, H. Hollerith developed a machine to sense (by electromechanical means) data on punched cards, and Hollerith's machines were used to sort and classify data from the 1890 United States census reports. However these efforts were neither of the scope nor complexity envisioned by Babbage.

By the 1930's the advances in electrical and mechanical engineering of the 20th century had made possible the realization of Babbage's ideas, and in 1939 work on a machine developed jointly by Howard Aiken of Harvard University and by engineers of the International Business Machines Corporation was started. This machine, the Automatic Sequence Controlled Calculator, or Mark I, was the first of the modern large-scale calculating machines and included all of Babbage's ideas and more besides. The Mark I was an electromechanical machine and very slow by today's standards, but in the years after 1944 it performed work the results of which have been used by most of the computing laboratories in the world today.

The first all-electronic machine, the ENIAC (Electronic Numerical Integrator and Calculator) was developed at the Moore School of Electrical Engineering of the University of Pennsylvania and was completed in 1946. This machine proved the feasibility of a really large electronic machine (it contained 18,000 vacuum tubes) and formed the basis for many future designs.

The major difference between the machines discussed so far and the present large-scale computing devices is the result of a concept developed, probably simultaneously, by groups in England and the United States. This concept is that of storing the program of instructions for the machine in the same manner as for the data on which the machine operates. This idea makes it possible for the program to cause the machine to make internal modification to the program while the machine is running, and this internally modifiable, stored-program concept has been in many ways the most revolutionary development in the history of machines. Prior to the development of this concept it was necessary to prepare a new program for each change in the kind of calculation to be performed. With the advent of modifiable stored programs it became possible to prepare a program which, when finished with one set of calculations, would cause the machine to modify the program to work on a new set of data and then to continue without human intervention until the required process was complete. The increase in flexibility in such a system over previous systems was so great as to be truly revolutionary.

The first machines to utilize fully the stored-program concept was the EDSAC (Electronic Delay Storage Automatic Computer) of the University Mathematical Laboratory, Cambridge, England; the Electronic Discrete Variable

Automatic Computer, or EDVAC, built at the University of Pennsylvania; and the Whirlwind I of the Massachusetts Institute of Technology. Other stored-program machines based on earlier designs were the Universal Automatic Computer (the famous UNIVAC) designed by Presper Eckert and John Mauchly, (who had been at the University of Pennsylvania and then at Remington Rand, now a division of the Sperry Rand Corporation); the Institute for Advanced Study Computer, IAS, developed under the direction of John von Neumann; and the International Business Machine Corporation 701 Calculator.

Most of the early machines built before 1955 differed from each other in many details even though they were basically similar in design, but more recent machines show marked similarities in that they generally possess several units of which only one is the computer or processor. The other units perform the operations of control, storage, and input or output of data. Since these units are available in many configurations depending on the nature of the application, it is more nearly correct to call such a configuration a computing system rather than a computer.

While the original computers were built to solve a relatively few but important scientific problems, the modern computing systems have been used to perform payroll calculations, prepare weather forecasts, translate English into Braille, make airline reservations, and calculate the orbits of satellites, to name just a few of the many uses to which these systems have been put.

The future applications of such information-processing systems is apparently limited only by the size and speed of the system and by the ingenuity of the people who program problems for the systems.

(NOTE. Those interested in a more detailed history of computers may refer to the article "Calculating Machines" by H. H. Goldstine in the *Encyclopaedia Brittanica*, Vol. 4, 1961.)

INDEX

Q

R

S

T

U

V

W

M
N.NEA

FROM:

ERIC FACILITY
SUITE 601
1735 EYE STREET, N. W.
WASHINGTON, D.C. 20006