DOCUMENT RESUME

ED 116 928 SE 019 968

AUTHOR	Kim, K. Ed.; And Others
TITLE	Base Numeration Systems and Introduction to Computer Programming.
INSTITUTION	Institute for Services to Education, Inc., Washington, D.C.
SPONS AGENCY	National Inst. of Education (DHEW), Washington, D.C.
BUREAU NO	BR-7-0867
PUB DATE	71
CONTRACT	OEC-0-8-070867-0001
NOTE	64p.; Appendix material from ED 084 936; Occasional marginal legibility
EDRS PRICE	MF-$0.76 HC-$3.32 Plus Postage
DESCRIPTORS	*College Mathematics; Computer Programs; *Computer Science Education; Guides; Higher Education; Instruction; *Instructional Materials; Mathematics Education; Number Systems; Programing; Secondary Education; *Secondary School Mathematics; *Teaching Guides
IDENTIFIERS	FORTRAN Language; Thirteen College Curriculum Program

ABSTRACT

This teaching guide is for the instructor of an introductory course in computer programming using FORTRAN language. Five FORTRAN programs are incorporated in this guide, which has been used as a FORTRAN IV SELF TEACHER. The base eight, base four, and base two concepts are integrated with FORTRAN computer programs, geoblock activities, and related exercises. Each statement of the first FORTRAN program is described in detail with suggested discussion questions and activities. (Subsequent programs are given without detail.) The FORTRAN programs included are: (1) change base eight numerals to base ten numerals, (2) determine the number of significant places for a given input data, (3) list the even numbers less than 200 for the base eight, (4) give the integral powers of ten in scientific notation, and (5) give the multiples of four in the base eight. Teaching suggestions include the modification of illustrated programs as well as activities for teaching of the design of a simple computer, unconditional and conditional transfer statements, and DO LOOPS. Fixed point (integer) system and floating point systems of computation in the digital computer are described. Problems with mathematical operation symbols complete the activities in the manual. (JBW)

BASE NUMERATION SYSTEMS AND INTRODUCTION TO COMPUTER PROGRAMMING

editorial committee

K. Kim (1970)
W. Mint (1970)
H. Hamilton (1970)
A. Lewis (1970)
L. Batra (1969-70)
B. Stubblefield (1969-70)
G. Latortue (1969-70)
M. Turner (1970)

THIRTEEN COLLEGE CURRICULUM PROGRAM
in conjunction with
CURRICULUM RESOURCES GROUP OF ISE

ABOUT THE INSTITUTE FOR SERVICES TO EDUCATION

The Institute for Services to Education was incorporated as a non-profit organization in 1965 and received a basic grant from the Carnegie Corporation of New York. The organization is founded on the principle that education today requires a fresh examination of what is worth teaching and how to teach it. ISE undertakes a variety of educational tasks, working cooperatively with other educational institutions, under grants from government agencies and private foundations. ISE is a catalyst for change. It does not just produce educational materials or techniques that are innovative; it develops, in cooperation with teachers and administrators, procedures for effective installation of successful materials and techniques in the colleges.

ISE is headed by Dr. Elias Blake, Jr., a former teacher and is staffed by college teachers with experience in working with disadvantaged youth and Black youth in educational settings both in predominantly Black and predominantly white colleges and schools.

ISE's Board of Directors consists of persons in the higher-education system with histories of involvement in curriculum change. The Board members are:

ABOUT THE THIRTEEN-COLLEGE CURRICULUM PROGRAM

From 1967 to the present, ISE has been working cooperatively with the Thirteen-College Consortium in developing the Thirteen-College Curriculum Program. The Thirteen-College Curriculum Program is an educational experiment that includes developing new curricular materials for the entire freshman year of college in the areas of English, mathematics, social science, physical science, and biology and two sophomore year courses, humanities and philosophy. The program is designed to reduce the attrition rate of entering freshmen through well thought-out, new curricular materials, new teaching styles, and new faculty arrangements for instruction. In addition, the program seeks to alter the educational pattern of the institutions involved by changing blocks of courses rather than by developing single courses. In this sense, the Thirteen-College Curriculum Program is viewed not only as a curriculum program with a consistent set of academic goals for the separate courses, but also as a vehicle to produce new and pertinent educational changes within the consortium institutions. At ISE, the program is directed by Dr. Frederick S. Humphries, Vice-President. The curricular developments for the specific courses and evaluation of the program are provided by the following persons:

Course	ISE STAFF
English	Miss Joan Murrell, Senior Program Associate Miss Carolyn Fitchett, Program Associate Mr. Sloan Williams, Program Associate Miss Tiveeda Williams, Research Assistant Miss Ernestine Brown, Secretary
Social Science	Dr. George King, Senior Program Associate Dr. William J. Massie, Jr., Program Associate Miss Nancy Seymour, Research Assistant Miss Gloria Lowe, Secretary
Mathematics	Dr. Beauregard Stubblefield, Senior Program Associate Mr. Bernis Barnes, Program Associate Dr. Phillip McNeil, Program Associate Dr. Walter Talbot, Consultant Mrs. Debrah Johnson, Secretary
Physical Science	Dr. Leroy Colquitt, Senior Program Associate Dr. Ralph Turner, Consultant Mrs. Cynthia Paige, Secretary
Biology	Dr. Charles Goolsby, Senior Program Associate Dr. Daniel Obasun, Program Associate Dr. Paul Brown, Consultant
Humanities	Mr. Clifford Johnson, Senior Program Associate Miss Marguerite Willett, Secretary
Philosophy	Dr. Conrad Snowden, Senior Program Associate Dr. Henry Olela, Program Associate Miss Valerie Simms, Program Associate Miss Judith Burke, Research Assistant Miss Faith Halper, Secretary
Evaluation	Dr. Thomas Parmeter, Senior Research Associate Dr. Joseph Turner, Senior Research Associate Mr. John Faxio, Research Assistant Mrs. Judith Rogers, Secretary

In addition, Miss Patricia Parrish serves as general editor of the curriculum materials as well as an Administrative Assistant to the Director. Mrs. Joan Cooke is Secretary to the Director.

The curriculum staff is assisted in the generation of new educational ideas and teaching strategies by teachers in the participating colleges and outside consultants. Each of the curriculum areas has its own advisory committee, with members drawn from distinguished scholars in the field but outside the program.

The number of colleges participating in the program has grown from the original thirteen of 1967 to nineteen in 1970. The original thirteen colleges are:

Alabama A and M University	Huntsville, Alabama
Bennett College	Greensboro, North Carolina
Clark College	Dallas, Texas
Florida A and M University	Atlanta, Georgia
Jackson State College	Tallahassee, Florida
Lincoln University	Lincoln University, Pennsylvania
Norfolk State College	Norfolk, Virginia
North Carolina A and T State University	Greensboro, North Carolina
Southern University	Baton Rouge, Louisiana
Talladega College	Talladega, Alabama
Tennessee State University	Nashville, Tennessee
Voorhees College	Denmark, South Carolina

A fourteenth college joined this consortium in 1968, although it is still called the Thirteen-College Consortium. The fourteenth member is:

Mary Holmes Junior College	West Point, Mississippi

In 1970, five more colleges joined the effort although linking up as a separate consortium. The members of the Five-College Consortium are:

Elizabeth City State University	Elizabeth City, North Carolina
Langston University	Langston, Oklahoma
Southern University at Shreveport	Shreveport, Louisiana
Saint Augustine's College	Raleigh, North Carolina
Texas Southern University	Houston, Texas

In 1971, eight more colleges joined the curriculum development effort as another consortium. The member schools of the Eight College Consortium are:

Alcorn A and M College	Lorman, Mississippi
Bethune-Cookman College	Daytona Beach, Florida
Grambling College	Grambling, Louisiana
Jarvis Christian College	Hawkins, Texas
LeMoyne-Owen College	Memphis, Tennessee
Southern University in New Orleans	New Orleans, Louisiana
University of Maryland, Eastern Shore	Princess Anne, Maryland
Virginia Union University	Richmond, Virginia

The Thirteen-College Curriculum Program has been supported by grants from:

The Office of Education, Title III, Division of College Support
The Office of Education, Bureau of Research
The National Science Foundation, Division of the Undergraduate Education
The Ford Foundation
The Carnegie Corporation
The Esso Foundation

STAFF AND TEACHERS OF THE MATHEMATICS PROGRAM

Four Years In The Program

Margaret Artis (NC)
Bernis Barnes (JA & ISE)
Laj Batra (CL)
Calvin Browne (TE & AL)
Reuben Drake (NC)
Carolyn Harris (CL)
Janie Jordan (NF)
Nancy Ledet (TE)
Johnsie Jo Posey (SB)
Carl Whitman (FL)

Three Years In The Program

Joseph Colen (JA)
Charles Haynie (ISE)
Dorothy Hogan (BI)
Roger Ingraham (VO & BE)
Carolyn C. Johnson (NF)
Harold King (BI)
James Kirkpatric (AL)

Two Years In The Program

Jack Alexander (ISE)
Ruth Carter (AL)
Beatrice Clarke (FL)
John Ernst (MH)
Lee Evans (ISE)
Prabhat Hazra (LI)
Kenneth Hoffman (TA)
Gwendolyn Humphrey (FL)
Reuben Kesler (VO)
Vivian Kline (LI)
Huber Lamotte (SB)
Gerard Latortue (VO)
Vernon Lowenber[illegible] (TA)
Theodore Morgan (SB)
A. K. Mukherjee (LI)
Dorothea Smith (AL)
Beauregard Stubblefield (ISE)

One Year In The Program

Douglas Anderson (JA)
William Barclay (ISE)
Barbara Bardwell (BI)
Hollie Baker (NF)
Joel Brainard (TA)
Willie Brown (JA)
Donald Butler (LA)
Nancy Everett (SS)
Robert Ghent (TA)
Henrilynn Gordon (TS)
Herbert Hamilton (TS)
Alfred Hawkins (BI)
Robert Hollister (TA)
John James (BE)
Ki W. Kim (LA)
Henry Lewis (SA)
Addessa Lewis (SA)
Mary Love (TE)
V. P. Manglick (EC)
Robert McKean (LI)
S. S. Sachdev (EC)
Eddie Paramore (BE)
Walter Talbot (ISE)
Melvin Turner (SS)
Elbert White (MH)
Frank Wyse (TA)

Summer

Fred Binford (TE)
Arthur Bragg (ISE)
Evelyn Edwards (ISE)
Boyd Jones (NF)
Phillip McNeil (ISE)
Virginia Merrill (ISE)
Harold Murray, Jr. (SAC)
Win [illegible]yint (TE)
William Nicholson (ISE)
Mary Payton (SAC)
Argelia Rodriguez (BI)
Jean Savary (BI)
Euclides Torress (BI)
Ray Treadway (BE)
John Wiley (ISE)

(AL)-Alabama A & M; (BE)-Bennett; (BI)-Bishop; (CL)-Clark; (EC)-Elizabeth City; (FL)-Florida A & M; (JA)-Jackson State; (LA)-Langston; (LI)-Lincoln; (MH)-Mary Holmes; (NF)-Norfolk State; (NC)-N. Carolina A & T; (SB)-Southern U.; (SS)-Southern at Shreveport; (SA)-St. Augustine; (TA)-Talledega; (TE)-Tennessee; (TS)-Texas Southern; (VO)-Voorhees; (SAC)-Southern Association; (ISE)-Institute For Services to Ed.

BASE NUMERATION SYSTEMS
AND
INTRODUCTION TO COMPUTER PROGRAMMING

Nearly everyone has heard of the electronic device known as the electronic computer which has been developed and is in widespread use by businessmen, scientists, and engineers to assist them in obtaining the solutions of many types of problems. A computer program is a collection of instructions (arranged in a predetermined order) which allows the computer to carry out a sequence of functions for the express purpose of obtaining the solution of a particular problem.

In this unit you will find many problems that the computer has helped to solve. It is only natural to expect that the computer programs used as well as the solutions will at times be included. Thus, this allows you to be told some ways to recognize problems you may have that can benefit from the method used Because very little extra computer time would be needed to solve other problems of the same style, these solutions are included as proper subsets of larger sets. Thus, the teacher may use this unit, not only to find the answers to those problems contained herein, but also to select from among the others, additional exercises and test questions.

This unit has been successfully used as a FORTRAN IV SELF-TEACHER. Each type of instruction listed in the program is described in detail before its use. The descriptions are given in the order in which the instructions appear in the unit. If any computer instruction has been previously described, it is so indicated by reference* either to a previous program or to an earlier portion of the current program. Also, sample data and the manipulation thereof are given in a form so that the reader may follow the results throughtout the program.

We shall use base numeration systems as bases from which a programming effort will be launched. The base of a numeration system is named according to the number of symbols used to represent numbers in that system. For example, there is only one symbol, 1, for the Base One Numeration System. For this system, we write

1	to represent the number of moons the earth has,
11	to represent the number of elements in a pair,
111	to represent the number of members of a trio, and
1111	to represent the number of people in a quartet.

We would use a string of a dozen 1's to represent the number of eggs in a carton. Base One System is equivalent to tallying. Note that we cannot write zero in this system. At the other extreme, in the Base Infinity System, there is a different symbol used for each number. Necessarily, there are infinitely many symbols as every number must be represented by a different symbol.

*The reference is given in columns 73-80 of the instruction. Columns 73-80 form the non-executable part of the instruction. As an example, the reference 2-7 means, "See Instruction #7 of Program #2.

In the Base One System, it would be a tedious task to write representations for large numbers, and the Base Infinity System would be too taxing on the brains to read. Therefore, we shall limit our discussion to finite base numeration systems which lie between these two systems.

2.1 Base Eight As A System of Numeration

We shall now experiment with a base numeration system which is between the Base One System and the Base Infinity System. In making use of our previous experiences in the Base Ten System, we experiment with the Base Eight System* of Numeration in which there are exactly eight symbols, 0, 1, 2, 3, 4, 5, 6 and 7. In this system

1 represents the number of noses one has,

2 represents the number of shoes a person wears,

3 represents the number of sides of a triangle,

4 represents the number of bases of a baseball diamond,

5 represents the number of toes on a normal right foot,

6 represents the number of sides of a hexagon,

7 represents the number of continents in the world,

10 represents the number of letters in the word computer**

11 represents the number of players on the starting baseball team,

*One way to write from 1 to 100 in Base Eight is to write from 1 to 100 in Base Ten, skipping any number having a numeral representation containing an 8 or a 9.

**Note: Verbal translation of 10 (Base Eight) is one zero rather than the word ten since ten means ten ones and 10 (Base Eight) means eight ones. Hence, careful verbal symbolization should be used to prevent ambiguity in the exact meaning of the symbol.

12 represents the number of fingers on your two hands,

13 represents the number of members of a football team,

14 represents the number of eggs in a dozen,

15 represents the unlucky number, and

16 represents the number of days in a fortnight.

Using the Base Eight Numeration System, give the numerals to represent:

(a) the number of pennies in three nickels

(b) the square of four

(c) the ninth odd number

(d) the ninth even number

(e) the tenth odd number

(f) the tenth even number

(g) the eleventh odd number

(h) the eleventh even number

(i) the twelfth odd number

(j) the number of hours in a day

(k) the square of five

(l) the number of letters in the alphabet

(m) the cube of three

(n) the number of days in February (non-leap year)

(o) the number of days in February in a leap year

(p) the number of days in September

(q) the number of days in January

(r) the fifth power of two

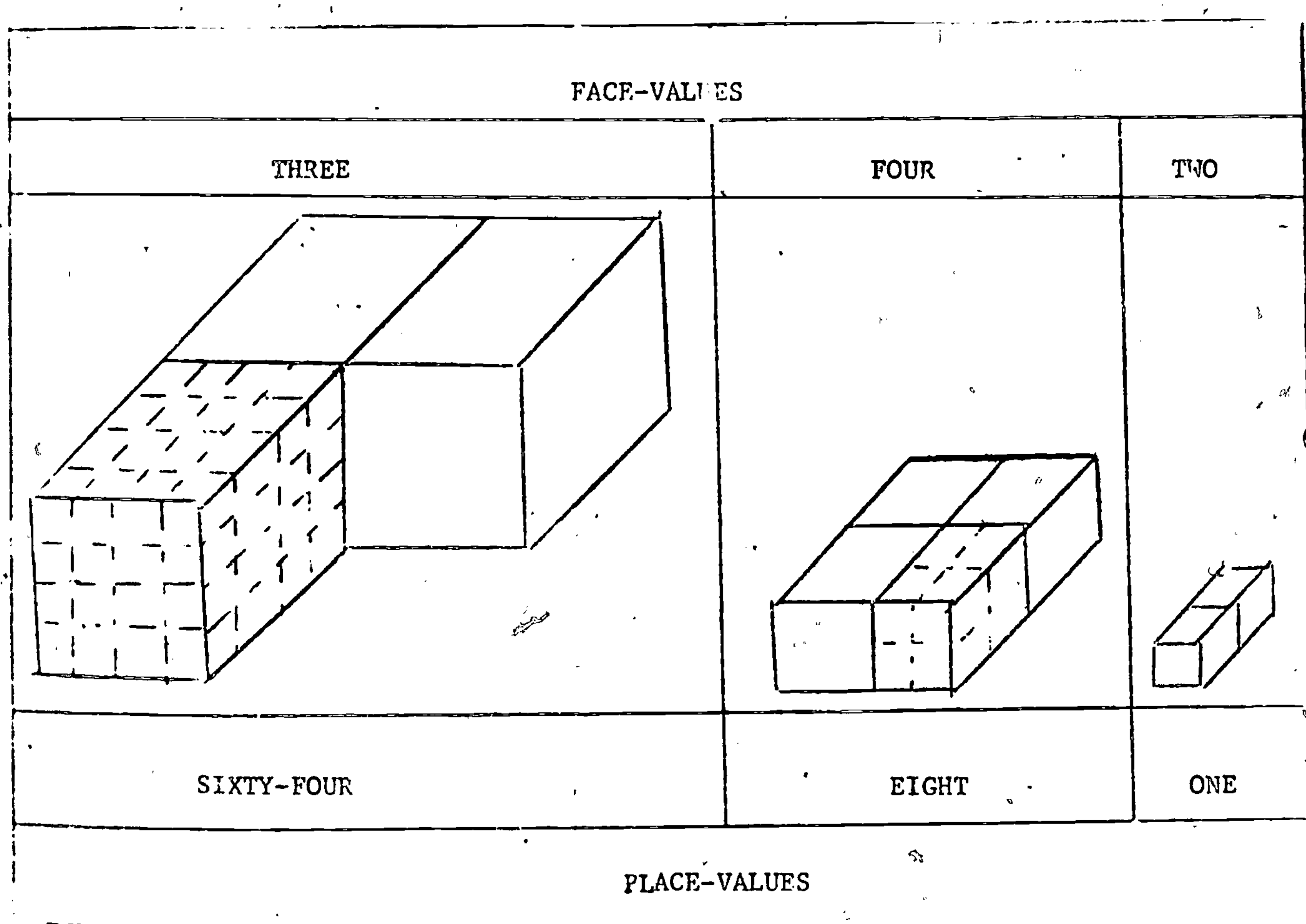
(s) the product three and eleven

(t) the number of days in five weeks

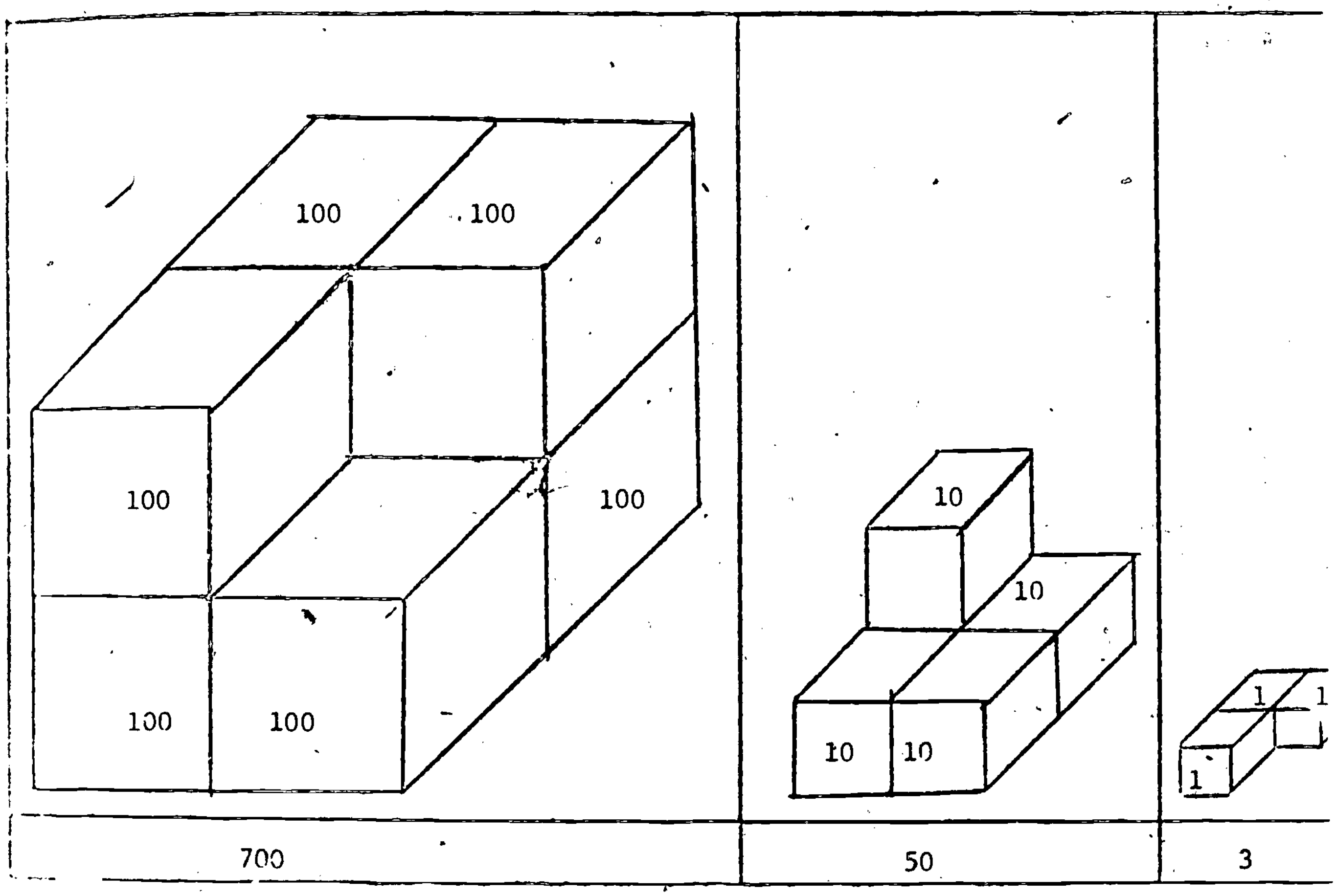
(u) the number of inches in a yard

2.2 Place-Values And Face-Values In The Base Eight Numeration System

In the Base Eight Numeration System numerals are written to indicate numbers. Each symbol of the numeral has two values, a face-value and a place-value. In the numeral 342 the face-value of 2 is two and the place-value of two is one. The face-value of 4 is four and the place-value of 4 is eight. The face-value of 3 is three and the place-value of 3 is sixty-four. The corresponding Base Ten numeral is given by 3 x 64 + 4 x 8 + 2 x 1 or 226.


FACE-VALUES
THREE
FOUR
TWO
SIXTY-FOUR
EIGHT
ONE
PLACE-VALUES

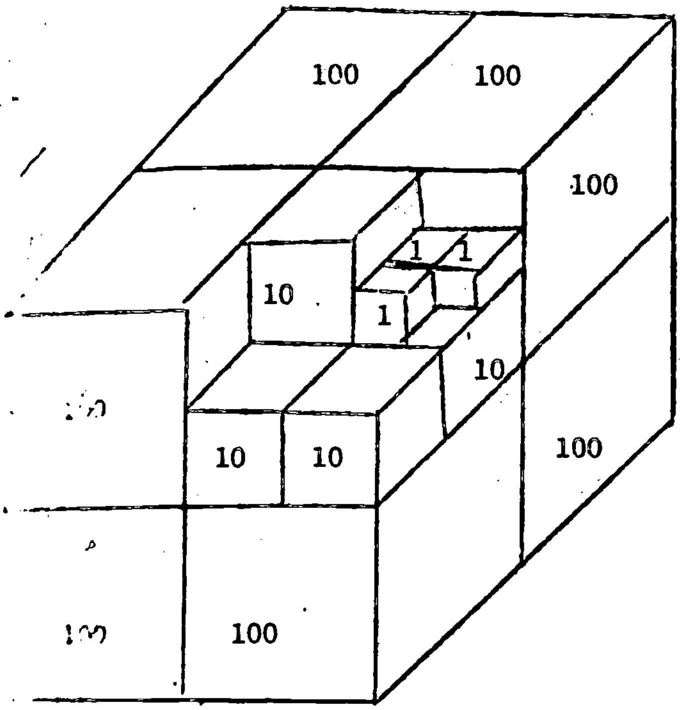
Example 1. State the place-value shown by each symbol in the Base Eight numeral 753.



The 7 in the numeral 753 is a symbol for seven eight-eights or seven sixty-fours.

The 5 in the numeral 753 is a symbol for five eights.

The 3 in the numeral 753 is a symbol for three ones.



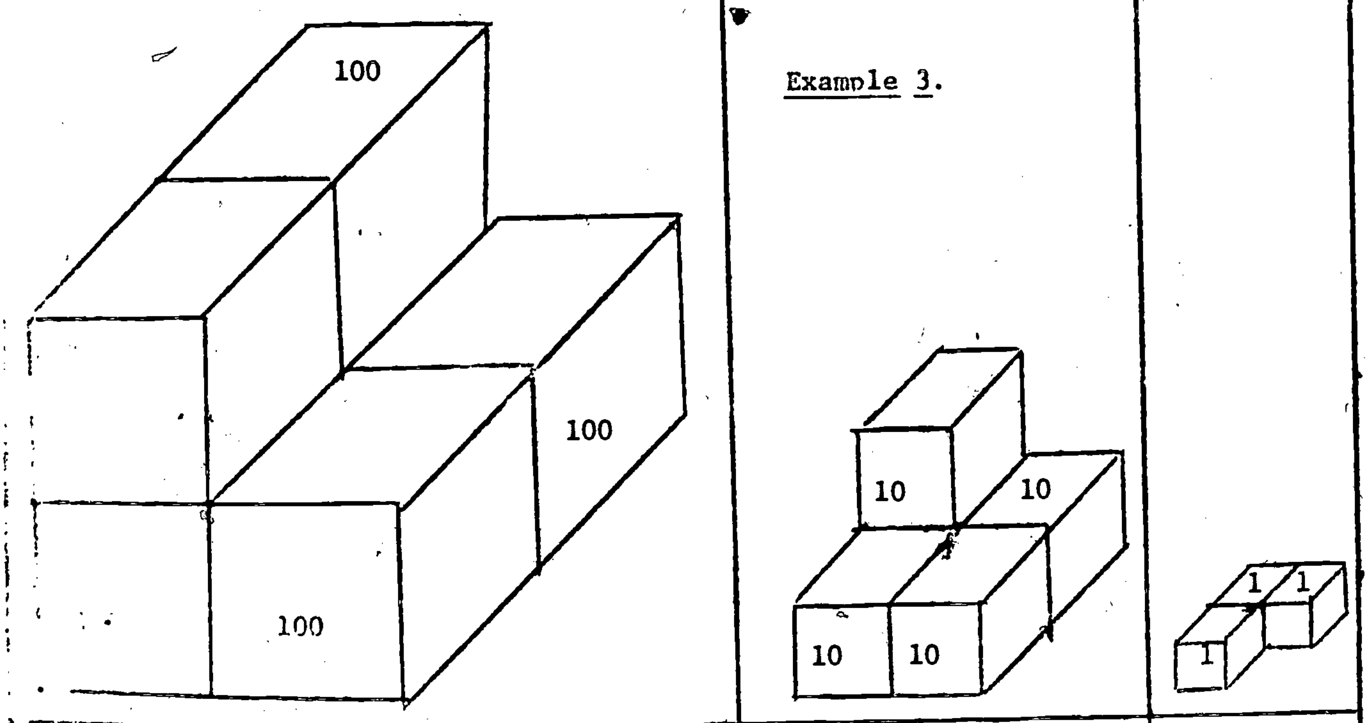
Example 2. How many blocks should be added to the set on the left to make an octet of octets of octets?

Answer. Five blocks are needed to go with the three ones to get an octet. Two more octets are needed to get eight octets. This octet of octets will be sufficient to complete the eight octets of octets.

Total. 2 blocks of 8 and 5 unit blocks which can be written as 25 in base 8.

Exercise: Find the face-values and place-values in the Base Eight numeral 653.

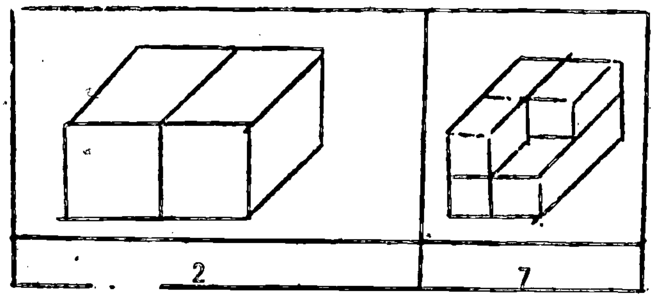
Example 3.



six sixty-fours | five eights | three ones

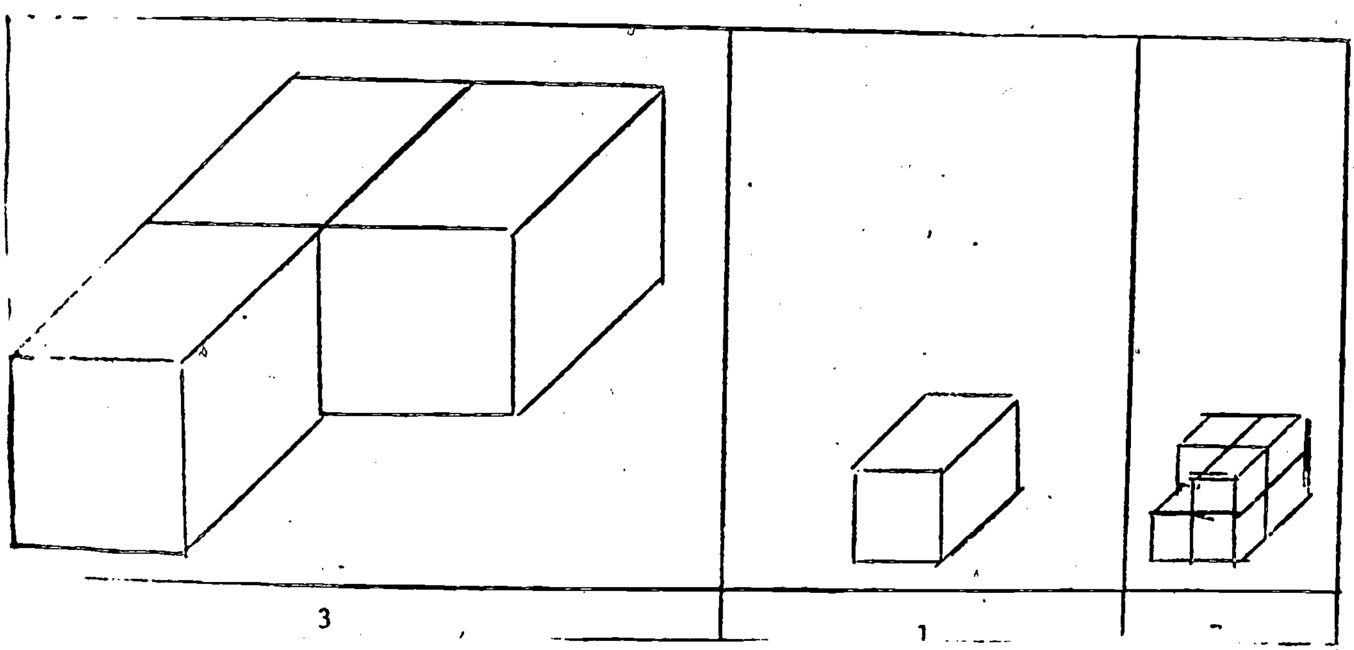
Example 4. Expand each of the following Base Eight numerals in Base Ten.

(a)



Answer:	Base Ten Face-Values	2	,	7	
		↑		↑	
	Base Eight Numeral	2		7	= 2 x 8 + 7 x 1 as Base Ten Numeral 23
		↓		↓	
	Base Ten Place-Value	8	,	1	

(b)



BASE TEN FACE-VALUES	3	,	1	,	7		
	↑		↑		↑		Base Ten Numeral
BASE EIGHT NUMERAL	3		1		7	=	3 x 64 + 1 x 8 + 7 x 1 = 207
	↓		↓		↓		
BASE TEN PLACE-VALUES	64	,	8	,	1		

Games:

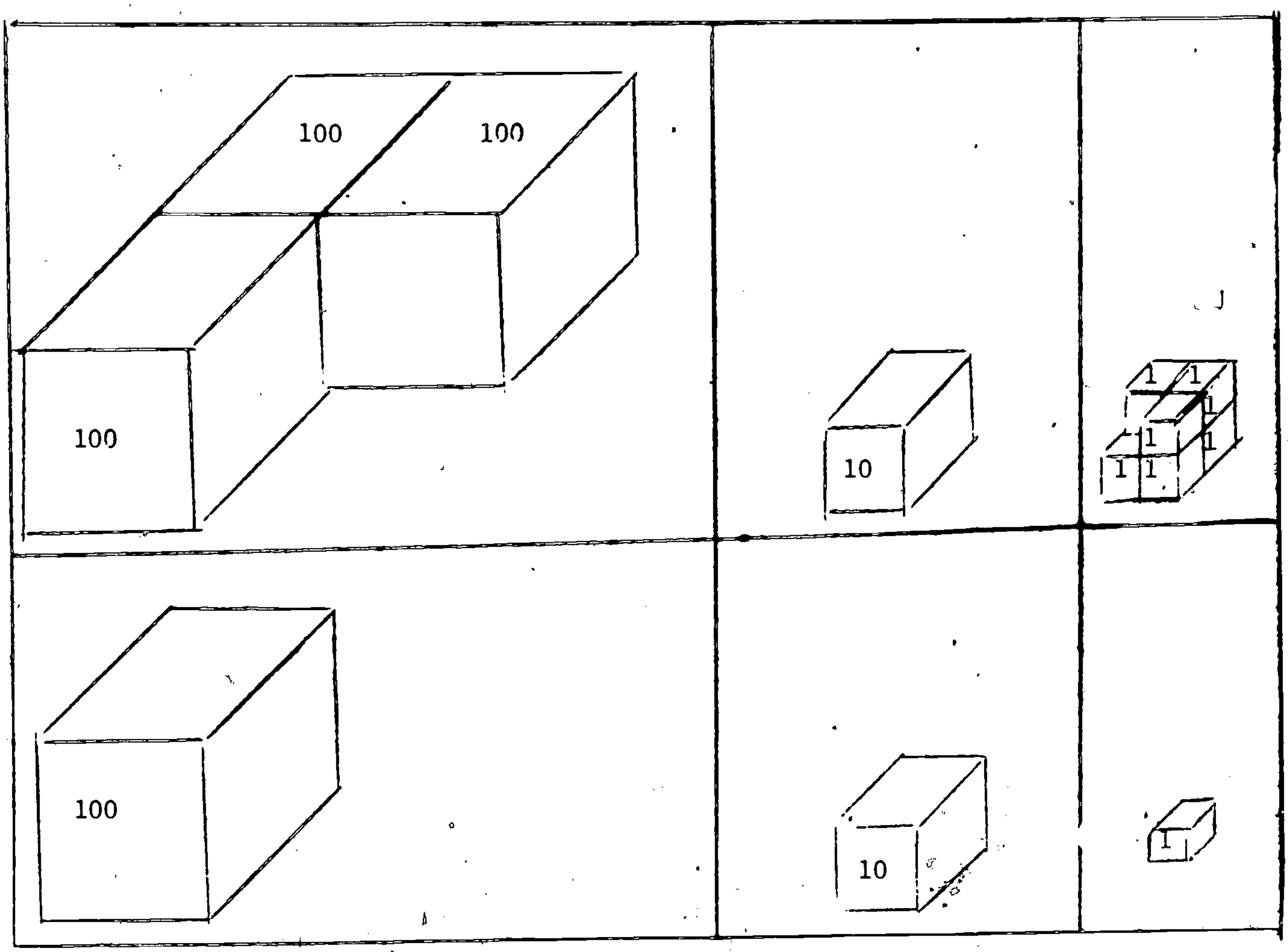
(a) Use cubic Geo-blocks to indicate the Base Eight numeral 33.

(b) Use cubic Geo-blocks to indicate the Base Eight numeral 47.

(c) Use cubic Geo-blocks to indicate the Base Eight numeral 77.

(d) Use cubic Geo-blocks to indicate the following addition:

317	327	777
+111	+222	+ 1



(e) Use cubic Geo-blocks to indicate the following subtraction.

446	543	333
-222	-234	-245

We see that the last symbol on the right of a numeral indicates the number of ones, the next to last symbol gives the number of base numbers, the next symbol gives the number of squares of base numbers, the next symbol gives the number of cubes of base numbers, etc. With this knowledge, we can easily convert a base eight numeral to a base ten numeral. In writing and reading numerals, we think of them as the numbers that they represent.

The number $13_8 \quad = \quad 3_{10} + 8_{10} = 11_{10}$

The number $4726_8 \quad = \quad 6_{10} + 2_{10}(8_{10}) + 7(8_{10})^2 + 4(8_{10})^3$

$\quad = \quad 2{,}518$

The number $125_8 \quad = \quad 5_{10} + 2_{10}(8_{10}) + 1(8_{10})^2 = 85_{10}$

Exercises

1. Complete the following table:

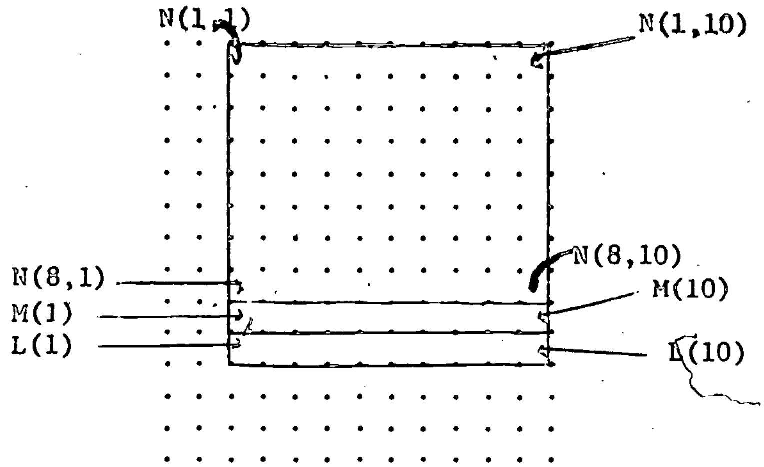
Base Eight		1	7	21		35	427
Base Ten	0				27		

* * *

FORTRAN PROGRAM NO. 1

```
                                                                          Referen.
1234567        20          30          40          50          60          70  Columns
C     BEAU GERA LAJ                                                        COMMENT
C     PROBLEM NO. 1 PAGE 40                                                COMMENT
C     PROGRAM TO CHANGE TEN EIGHT-PLACE BASE EIGHT NUMBERS TO BASE TEN     COMMENT
C     SPACE CALLED NCRDS TELLS US THE NUMBER OF CARDS OF DATA (2 DIGITS)   COMMENT
C     WE SHALL RESERVE EIGHTY SPACES FOR THE RECTANGULAR ARRAY N(8,10)     COMMENT
C     WE SHALL RESERVE TEN SPACES FOR THE BASE TEN ANSWERS M(10)           COMMENT
C     WE SHALL RESERVE TEN SPACES FOR THE BASE EIGHT INPUT NUMBERS L(10)   COMMENT
    1 DIMENSION N(8,10), M(10), L(10)
    2 READ (1,3) NCRDS
    3 FORMAT (I2)
    4 DO 24 K = 1, NCRDS
C     THE FOLLOWING READ STATEMENT CAUSES DATA TO BE STORED BY COLUMNS.    COMMENT
    5 READ (1,6) N                                                         1-2
    6 FORMAT (80I1)                                                        1-3
    7 DO 21 J = 1, 10                                                      1-4
    8 MX = 0
    9 LX = 0                                                               1-8
   10 FORMAT (1H , 10I10)
   11 DO 18 I = 1, 8                                                       1-4
   12 IF (I . EQ . 8) GO TO 16
   13 MX = MX + N(I,J) * 8**(8-I)
   14 LX = LX  + N(I,J)* 10**(8-I)                                         1-13
   15 GO TO 18
   16 MX = MX + N(I,J)                                                     1-13
   17 LX = LX + N(I,J)                                                     1-13
   18 CONTINUE
   19 M(J) = MX                                                            1-13
   20 L(J) = LX                                                            1-13
   21 CONTINUE                                                             1-18
   22 WRITE (3,10) L
   23 WRITE (3,10) M                                                       1-22
   24 CONTINUE                                                             1-18
   25 STOP
   26 END
```

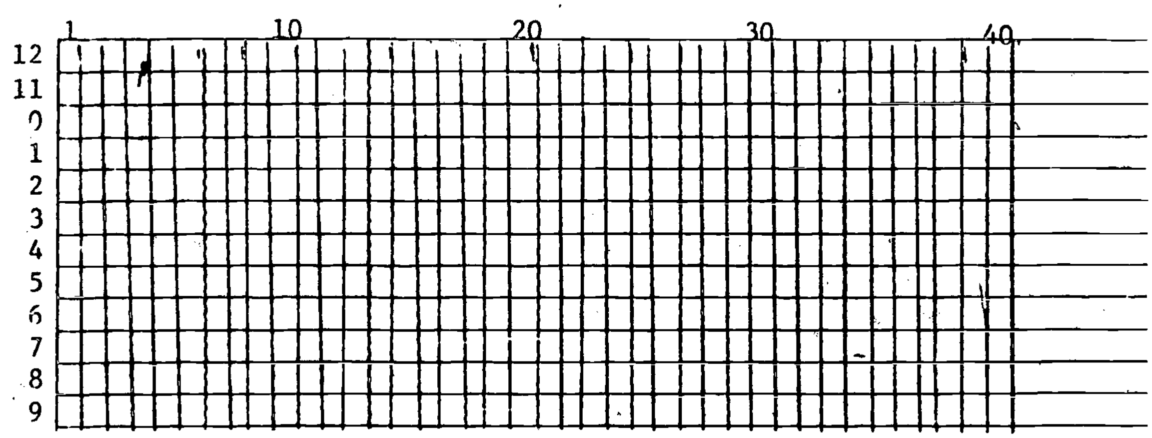
Use the Geo-board to follow the given data throughout the execution of the instructions in the FORTRAN PROGRAM NO. 1. After the execution of the first program instruction, the Geo-board may look like this:



EXPLANATION OF FORTRAN PROGRAM NO. 1

Each line in the program represents a program card. There are eighty columns on each card. In each column there is either a blank space or one of less than 50 symbols represented. The permissible symbols are the ten numerals (from 0 to 9), the twenty-six letters of the alphabet (from A to Z), the period, the comma, the left parenthesis, the right parenthesis, the equal sign, the plus sign, the minus sign, the asterisk and the dollar sign.

Each column of an eighty column card has twelve slots which are labeled as shown below.



We may punch holes in these slots. Depending upon which particular column slots are punched in a column, we represent various symbols. Since there are twelve slots in each column, there are 2^{12} possible combinations of punched slots for each of the columns. However, since there are fewer than 50 symbols to be used, (10 numbers, 26 letters, etc.), then the 2^{12} different combinations are more than adequate.

Each card will have either a C representation in the first column or will have nothing there. If a card has a C in the first column, then the information is given to the processor that the information on the card is <u>only</u> a <u>C</u>omment. That is, the information is not to be processed. Another way of saying this is that the processor will ignore these comments.

We note that in the program under study, the first seven cards are comment cards.

Question: Is there another comment card in this program? If so, what is it?

If a card does not contain a C in the first column, then the card contains at least in part, <u>a program statement</u>, and this program statement may in turn be either <u>executable</u> or <u>not executable</u>. Executable statements are executed in the order that they are listed unless changed by a control statement.

<u>First Statement</u>

DIMENSION N(8,10) , M(10), L(10)

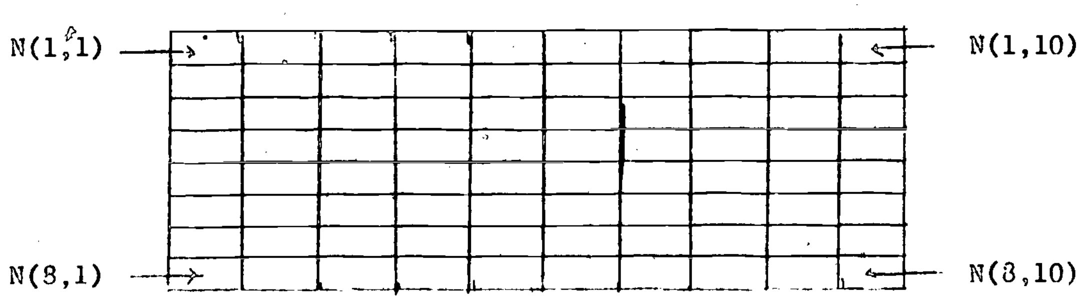
This first statement is called a DIMENSION STATEMENT.

The first program statement is an executable statement: Since all numbered statements must be numbered in columns 2 to 5, and this statement is numbered, its number, "1" is placed in column 5. However, since this statement is never referred to in the program by another statement, it need not have been numbered. Also, though numbered, it need not have been numbered "1". Any number would have sufficed.

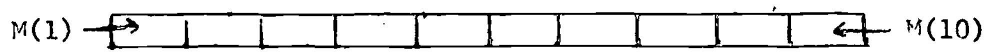
Numbered statements do not have to be listed in numerical order.

This first statement requests the processor to reserve space in the storage compartment of the computer for <u>three separate arrays of numbers</u>.

(a) An 8 x 10 two-dimensional array as shown below,



(b) A one-dimensional array of 10 spaces as shown below,



(c) Another one-dimensional array of 10 spaces.

Note: All statements begin in column #7.

Draw a picture representation for the array mentioned in (c) above. We should emphasize that the names of the spaces to be reserved

(a) may not be more than six characters long, and
(b) must consist of letters, or letters and numerals, the first of which must be a letter.

Examples:

The name	NAME	is acceptable.
The name	RESERVE	is not acceptable. (too many characters)
The name	2MR	is not acceptable. (a number comes first.)
The name	MR2	is acceptable.
The name	MR,2	is not acceptable.(The comma is neither a letter nor a numeral.)

In a DIMENSION STATEMENT, all names of spaces to be reserved are to be subscripted. However, subscripts are not to be counted when determining the length of the name.

Example

The name BOY(1,1) is acceptable. (There are only the characters B, O, and Y in the name inself.)

Questions:

(a) What is the total number of spaces that are reserved by the first statement?

(b) Can you write a statement which causes the processor to reserve a total of two hundred fifty spaces? If so, what is it?

(c) Can a subscript have more than six characters? If so, give an example.

(d) Can a statement have a larger statement number than the one which follows it?

(e) How large can a statement number be?

Second Statement

READ (1,3) NCRDS

This statement is called a READ STATEMENT. It is an executable statement; it requests the processor

(a) to reserve one space in the storage compartment of the computer which hereafter will be referred to as NCRDS if such named space has not previously been reserved (In case that such space has previously been reserved, this request is to be ignored.), and

(b) to read from machine number one (THE CARD READER) into this space (called NCRDS), the number which is on the next card in machine number one. read according to information FORMAT NO. 3. Thus Statement No. 3 tells the processor how the number appears on the card.

(c) to ignore all empty spaces as always.

Third Statement

FORMAT (I2)

This statement is called a FORMAT STATEMENT. A format statement is not an executable statement. This statement

(a) is numbered since it is referred to in the program; (Remember that since a FORMAT STATEMENT is not executable, to be used in a program, it must be referred to; hence it must be numbered.),

(b) is referred to by the second statement, and

(c) tells the processor that the number to be read according to the second statement is an integer and occupies the first two columns on the card to be read.

FORMAT STATEMENTS may or may not follow the statement which refers to it; such statements may be placed any where in the program.

Fourth Statement

DO 24 K = 1, NCRDS

The fourth statement is called a DO STATEMENT. It asks the processor to execute all statements following this DO STATEMENT down to and including statement number 24 as many times as the number in the computer storage space called NCRDS indicates. (For example, if NCRDS contains the number 9, then the statements following this DO STATEMENT down to statement number 24 will be executed in order 9 times.) However, before these statements are executed the first time, a space called K is reserved in the computer by the processor and is given the initial value of 1. The number in space K is increased by 1 each time the sequence of numbers has been executed.

Questions: Is the second statement numbered? If so, could this number have been omitted? What about the third and fourth statements?

Interpret the fifth, sixth, and seventh statements.

Eighth Statement*

MX = 0

This statement is called a SUBSTITUTION STATEMENT. It has two functions:

(a) To request the processor to reserve in the storage compartment of the computer, a space called MX, if such space has not already been reserved, and

(b) to put the value of the right hand expression in that space.

Tenth Statement

FORMAT (1H , 10I10)

The tenth statement gives two bits of information to the processor:

(a) It tells the processor to print, using the printer, as many characters after the H as the number before the H indicates. (In the case of this function, blank spaces are regarded as characters.)

(b) It tells the processor the form that the 10 numbers should take whenever this format statement is referred to. In this particular case, statements 22 and 23 both refer to the statement number 10. Each of the ten numbers is considered to be a ten place integer.

Questions:
(a) Must the eighth statement have been numbered?

*Note: The right hand expression need not have been equal to zero. It could have been either any other constant or any other type of algebraic expression. (See 1-13 also.)

(b) Is storage space reserved in the computer by the processor in the execution of statement number eight?

(c) What is the value of the expression on the right hand side of statement number eight?

(d) Give your interpretation of Statement number nine. Is storage space reserved in the computer by the processor in the execution of statement number nine?

(e) What is the value of the expression on the right hand side of statement number 9?

(f) What space is there reserved in this computer by this statement?

(g) Must the ninth statement have been numbered?

(h) Must the tenth statement have been numbered?

(i) What is your interpretation of Statement number eleven?

Twelfth Statement

```
If ( I . EQ . 8) GO TO 16
```

The twelfth statement is called a CONDITIONAL GO STATEMENT. It tells the processor to proceed to statement number 16 if the value of the number in space called I is equal to 8.

Thirteenth Statement

```
MX = MX + N(I,J) * 8**(8-I)
```

The thirteenth statement is another SUBSTITUTION STATEMENT. In determining the value of the expression on the right, the single asterisk is treated as a "times" sign and the double asterisk is treated as an "exponential" sign. As in all forms of mathematics, multiplication and division take presidence over addition and subtraction. Also, all forms of grouping have higher priority. In the case of computer programming, the exponentiation comes between multiplication and division, and grouping.

Questions:

(a) What is your interpretaiton of Statement Number 13?

(b) State the purpose of Statement Number 14.

(c) Suppose that the original data were Base Seven numbers.

*Statements similar to Statement 12 are given below:

(a) IF(I·NE·8)GO TO 16 means "If the value of I is not 8, transfer to S16".
(b) IF(I·LT·5)GO TO 7 means "Change control to S7 if I is less than 5."
(c) IF(J·GT·2)GO TO 3 means "Change control to S3 if J is greater than 2."

What changes would you make in Statement Number 13 in order to change those numbers to Base Ten?

(d) For Base Seven numbers, would you make any changes in Statement Number 14?

(e) Would Format Statement Number 10 have to be changed to accomodate Base Seven numbers?

(f) In rewriting FORTRAN PROGRAM NO. 1, would you make any other changes in this program if you wished to change ten eight-place Base Eight numbers to Base Ten?

Fifteenth Statement

GO TO 18

This statement is called an UNCONDITIONAL GO STATEMENT. It tells the processor to transfer control to statement number 18 under any condition.

Questions:

(a) Explain Statement Number 16.

(b) Explain Statement Number 17.

Note Statements 16 and 17 are to be executed only when I is equal to 8. At that time, statements 13 and 14 are not executed.

(c) Examine statements 13 and 14 to determine why the instructions were so programmed.

Note: The statements from Number 5 to Number 24 form our first DO LOOP. The statements from Number 8 through Number 21 form our second DO LOOP. Statements from Number 12 to Number 18 form our third DO LOOP. In any program, DO LOOPS may be nested: they must not be overlapping. In FORTRAN PROGRAM NO. 1, the three DO LOOPS are nested.

Eighteenth Statement

CONTINUE

The CONTINUE STATEMENT tells the processor to continue doing whatever you have started. It is sometimes desirable to have this statement as a reference point for a DO STATEMENT, especially when otherwise there might be two possible places that you may want a particular DO LOOP to end.

Questions:
(a) Explain Statement No. 19.
(b) Explain Statement No. 20.
(c) Explain Statement No. 21.

Twenty-Second Statement

WRITE (3,10) L

This is called a WRITE STATEMENT. It requests the processor to write all numbers in the one-dimensional array called L, using machine number 3 (the printer) according to FORMAT NO. 10. All leading zeros of a number will not be printed. Format statements must include an allowance of one extra space for the sign of a number.

Twenty-fifth Statement

STOP

The STOP STATEMENT tells the processor to execute no more statements.

Twenty-sixth Statement

END

The END STATEMENT tells the processor that this is the end of the program.

Problem. :

Using the data cards having input numbers arranged in the order indicated below, tell what happens when FORTRAN Program No. 1 is executed.

Card 1, 03

Card 2 06735627354673254362766352701734660513652371005673421634

Card 3 74556321540654310967801046014421473021364773621472221327

Card 4 1367213652112650366244513623106401263011 2[illegible]56214725137210

*

2. Complete the table below giving the number of places for the numbers listed.

Number	37	1000	657	7777	0004	1007
Places	two					

* * *

FORTRAN PROGRAM NO 2

```
1234567                                                          Reference
                                                                 Column
C      BEAU                                                      COMMENT
C      PROGRAM TO GIVE THE NUMBER OF SIGNIFICANT PLACES          COMMENT
       WRITE (3,22)                                              1-22
    22 FORMAT (2X, 37H PROGRAM 2 ***** PROBLEM 3 PAGE 41 //)     1-10
       DIMENSION BASE (10)                                       1- 1
       READ (1,1) BASE                                           1- 2
     1 FORMAT (10 F8.0)
       DO 20 I = 1, 10                                           1- 4
       B = BASE (I)                                              2- 1
       K = 8                                                     1- 8
     3 J = B/10.**K
       IF (J, NE, 0) GO TO 18                                    1-12
       K = K - 1                                                 1-13
       KI = K + 1                                                1-13
       GO TO 3                                                   1-15
    18 WRITE (3, 19)I, BASE (I), KI                              1-22
    19 FORMAT (I 20, F20.0, I20//)                               2-22
    20 CONTINUE                                                  1-18
       STOP                                                      1-25
       END                                                       1-26
```

(a) Fully describe the statement

WRITE (3,22)

(b) Each symbol / at the beginning or end of the Format detail causes the carriage return to become activated one extra time. When n is an integer, then the symbol nX in a Format statement activates the printer by causing it to skip n spaces. With this explanation, describe fully Statement No. 22.

(c) Describe the statement

DIMENSION BASE (10)

(d) Describe the statement

READ (1,1) BASE.

(e) The Format statement n F 8.0 tells the processor that there are n numbers being considered. Each one is a real number written in decimal form (or floating Point form). Each number is eight places long and there are zero places to the right of the decimal point. With this knowledge, tell what Statement No. 1 does.

NOTE: Heretofore, the only numbers we have been using have been integers. Names of variables which are integers must begin with one of the letters I, J, K, L, M and N. Other real numbers (with decimals) begin with letters other than these.

(f) Describe the following three statements.

DO 20 I = 1, 10

B = BASE(I)

K = 8

(g) The symbol in B/10. is self-explanatory. The number 10. is considered to be a real number having a decimal point. Tell what Statement No. 3 does. (Hint: J is an integer.)

(h) Tell what Statement 18 does.

(i) Tell what Statement 19 does.

(j) Tell what Statement 20 does.

3. The first six even whole numbers are represented in Base Eight by 0, 2, 4, 6, 10 and 12. List the first twenty even numbers given in Base Eight. How can you tell whether or not a whole number represented in Base Eight is even?

FORTRAN PROGRAM NO. 3

```
1234567
C     BEAU
C     PROGRAM TO LIST THE EVEN NUMBERS <200 IN BASE EIGHT
      WRITE (3, 22)'
   22 FORMAT (2X, 34H PROGRAM 3)
      DIMENSION N(4), M(4)
      M(1) = -8
      M(2) = -6
      M(3) = -4
      M(4) = -2
      DO 40 I = 1,25
      DO 30 J = 1,4
      M(J) = M(J) + 8
      N(J) = M(J)
   30 CONTINUE
      DO 20 J = 1,4
      K1 = N(J)/64
      R1 = M(J)-(K1*64)
      K2 = R1/8
      K3 = R1 - (K2*8)
      N(J) = (K1*100) + (K2*10) + K3
   20 CONTINUE
      WRITE (3,21) N
   21 FORMAT (4 I20 //)
   40 CONTINUE
      STOP
      END
```

Exercises

(a) Describe Statement No. 22.
(b) Describe Statement No. 21.

(c) Explain the statement

K1 = N(J)/64

(Hint: Since the name K1 begins with the letter K, K1 must be an integer.)

(d) Explain the statement

K2 = R1/8

*

4. In Base Ten the number <u>ten</u> raised to the second power is represented by 100 or 10^2; the number <u>ten</u> raised to the third power is represented by 1000 or 10^3. Give alternative representations for the following as integral powers of 10 in Base Eight.

(a) 100000 (c) 1000000

(b) 100000 (d) 10000000

***

<u>FORTRAN PROGRAM NO. 4</u>

```
1234567
C     BEAU
C     PROGRAM TO GIVE THE INTEGRAL POWERS OF 10 IN SCIENTIFIC NOTATION
    2 WRITE (3,22) NUMBERS < 10" USING ONE DIGIT TO THE LEFT OF THE DECIMAL
   22 FORMAT  (14X, 9H PROGRAM 4)
    1 DIMENSION T(8)
    5 READ (1,23)  T
   23 FORMAT (8I10)
    6 DO 20 I = 1,8
    7 A = T(I)
    8 K = 10
    3 J = A/10. ** K
    4 IF (J . NE .0)  GO TO 18
    9 K = K - 1
   10 GO TO 3
   18 WRITE (3, 19)   A, K
   19 FORMAT (12H THE NUMBER , F10.0  , 20H IS 10 TO THE POWER , I2)
   20 CONTINUE
   21 STOP
   22 END
```

Detailed Explanation of FORTRAN PROGRAM NO. 4

STATEMENT NUMBER	EXPLANATION
2	Printer Writes: PROGRAM 4*****PROBLEM 9 PAGE 42
1	Processor Reserves: [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] T(1) T(2) T(3) T(4) T(5) T(6) T(7) T(8)
5	Reader Reads In: [10.] [100] [1000] [10000] [100000] [1000000] [etc.] [etc.] T(1) T(2) T(3) T(4) T(5) T(6) T(7) T(8)
7	Space A is reserved and the value of T(1) is put there. [10] A
8	Space K is reserved and the value 10 is put there. [10] K
3	NOTE· 10. is a FLOATING POINT NUMBER. It is first raised to the power indicated by K. Then 10 is divided by the result and the decimal part is dropped. Hence the value of J is set to zero. [0] J
4	Statement No. 4 is self-explanatory if NE is read "IS NOT EQUAL TO".
9	Reduce the value of the number in space K by 1. [9] K
10	Change Control to Statement No. 3
18	After going through the above LOOP nine times, the number 1 is in space K and space A contains the number 10. Why? After the nineth time through, the PRINTER writes: THE NUMBER 10 IS 10 TO THE POWER 1.

*

5. The first five whole numbers that are multiples of four are represented in Base Eight by 0, 4, 10, 14, and 20.

   (a) Use cubic Geo-blocks to represent the number 4 in Base Eight.

   (b) Use cubic Geo-blocks to represent the number 10 in Base Eight.

   (c) Use cubic Geo-blocks to represent all of the multiples of four between zero and thirty-six.

   (d) Tell which of the following Base Eight numerals represent numbers that are multiples of four.

4402	10000	10000000	240000000004
4304	34005	32400000	200007655702
2067	20000	2567300	287565500400

   (e) Give a rule for telling whether or not a Base Eight numeral represents a multiple of four.

6. A number which is not even is ____. Two does not divide it. The first five odd whole numbers are represented in Base Four by 1, 3, 11, 13, and 21. List the first ten odd numbers given in Base Four. How can you tell whether or not a numeral indicates a Base Four odd number?

* * *

FORTRAN PROGRAM NO. 5

```
1234567
C     BEAU
C     PROGRAM TO GIVE THE MULTIPLES OF FOUR IN BASE EIGHT
   22 FORMAT (2X,35H PROGRAM 5***** PROBLEM 12 PAGE 44//)
      WRITE (3,22)
      DIMENSION A(100)
      INTEGER A, D4, D3, D2, D1, D5
      READ (1,1)-N
    1 FORMAT (I3)
      DO 3 I = 1,N
C THE FIRST 100 MULTIPLES OF FOUR ARE COMPUTED IN BASE TEN
      A(I) = I*4
C THE DIGIT ASSOCIATED WITH HUNDREDS PLACE IN BASE EIGHT IS COMPUTED
      D3 = A(I)/ 64
C IF D3 IS EQUAL TO ZERO, THEN THE STATEMENT LABELED 6 IS EXECUTED
      IF (D3.EQ.0) GO TO 6
C THE VALUE ASSOCIATED WITH D3 IS SUBTRACTED FROM THE MULTIPLE OF FOUR
      A(I) = A(I) - (D3*64)
C THIS STATEMENT COMPUTES THE DIGIT ASSOCIATED WITH THE TENS PLACE IN
C BASE EIGHT
    6 D2 = A(I)/8
C IF D2 IS EQUAL TO ZERO, THEN THE STATEMENT LABELED 8 IS EXECUTED
      IF (D2.EQ.0) GO TO 8
C THE VALUE ASSOCIATED WITH D2 IS SUBTRACTED FROM THE MULTIPLE OF FOUR
      A(I) = A(I) - (D2*8)
C THE REMAINING VALUE OF A(I) IS ASSIGNED TO THE DIGIT IN THE UNITS
    8 D1 = A(I)
C THE VARIOUS DIGITS ARE COMBINED INTO A THREE PLACE BASE EIGHT NUMBER
      A(I) = D3*100 + D2*10 + D1
    3 CONTINUE
      WRITE (3,10) (A(I), I = 1,N)
C     DISCUSS THE FOLLOWING FORMAT STATEMENT
   10 FORMAT (10X,10(I6,5X))
      STOP
      END
```

FORTRAN PROGRAM NO. 5 deserves further comments.

(a) The statement

INTEGER A, D4, D3, D1, D5

tells the PROCESSOR to round off these variables downward to the nearest integer and to treat them as integers.

(b) The statement

IF (D5 . EQ . 0) GO TO 2

is self-explanatory if the symbols EQ are interpreted together as "IS EQUAL TO" .

(c) The statement

WRITE (3,10) (A(I), I = 1,N)

causes the PRINTER to print all the numbers A(I) where I takes on all the integral values from 1 to the value of N.

(d) The 10(I6, 5X) in FORMAT NO. 10 has the same effect as if I6,5X had been written ten times in succession.

Questions:

(a) What does the 2X in FORMAT STATEMENT NO. 22 tell the PROCESSOR?

(b) What do the two symbols // in FORMAT STATEMENT NO. 22 tell the PROCESSOR?

(c) Explain in detail, the statement

WRITE (3,22)

in FORTRAN PROGRAM NO. 5.

(d) Explain in detail,

FORMAT NO. 10.

*

2.3 Base Four As A System of Numeration

In Base Four, we use the symbols 0, 1, 2, 3 and the numerals

0 to represent the number of whole nickels in a penny.

1 to represent the number of moons the earth has.

2 to represent the number of halves in a whole.

3 to represent the number of angles in a triangle.

10 to represent the number of seasons in a year.

11 to represent the number of people in a set of quintuplets.

12 to represent the number of eggs in a half used carton.

13 to represent the number of days in a week.

Exercises

1. Using the Base Four Numeration System, give numerals to represent

   (a) the number of toes on your feet

   (b) the number of weeks in a year

   (c) the number of pints in a gallon

2.4 Place Value and Face Value In The Base Four Numeration System

There are four subdivisions in each section and a builder constructs four sections each season of the year. There are four blocks in each quarter subdivision. The builder buys his 'one coat' wall paint in wholesale quantities at the price of four dollars per gallon. A case of wall paint consists of four gallons. A car basket of this paint consists of four cases. A forklift consists of four car baskets. A truckload consists of four forklifts. A shipment consists of four truckloads. A warehouse of wall paint consists of four shipments.

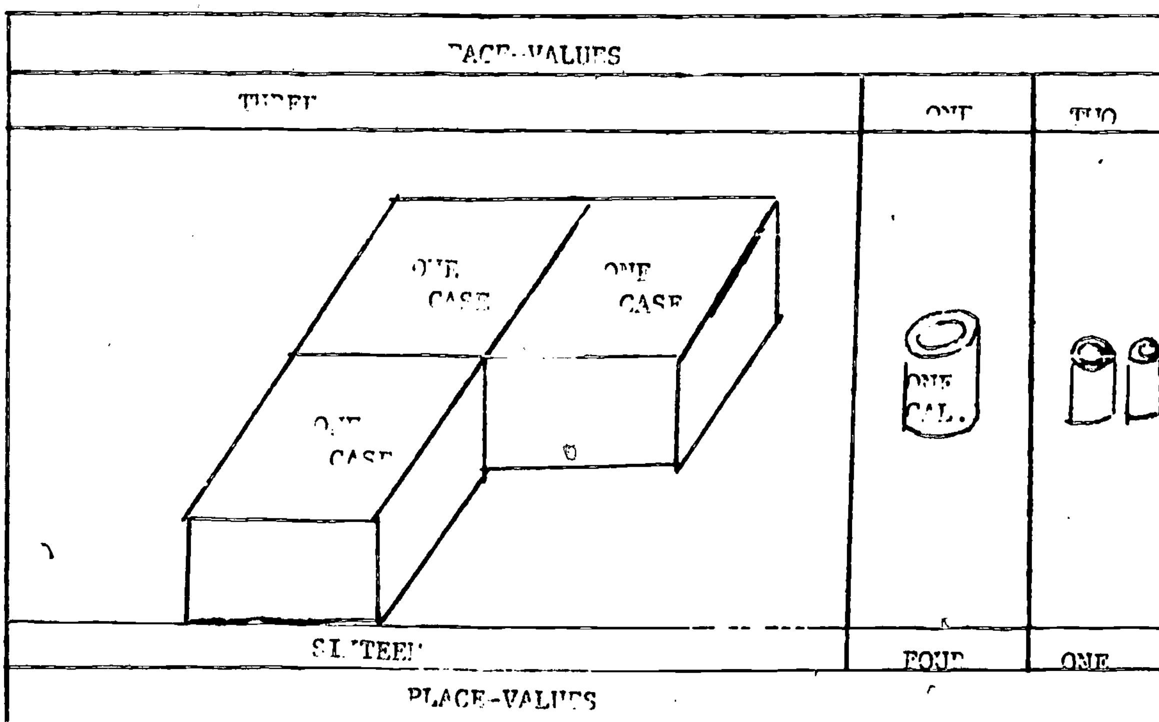
The builder constructs eight buildings on each side of the street in each block. Each building has four units each consisting of four four-room apartments. Since, on the average, a gallon of wall paint is used for each room, then a case of wall paint is used for each unit, and a forklift of wall paint is used for each building. A truckload of wall paint is used in each quarter-block and a shipment of wall paint is used in each block. A warehouse of wall paint is used in each quarter-subdivision and four warehouses of wall paint are used in each subdivision.

Obviously, a quart of paint is a wall of paint and a glass (or a cup) of paint is a quarter-wall of paint. A glass of paint is used in each pan and a half gill of paint is considered to be one roller of paint. We have <u>THE PAINT MODEL</u>

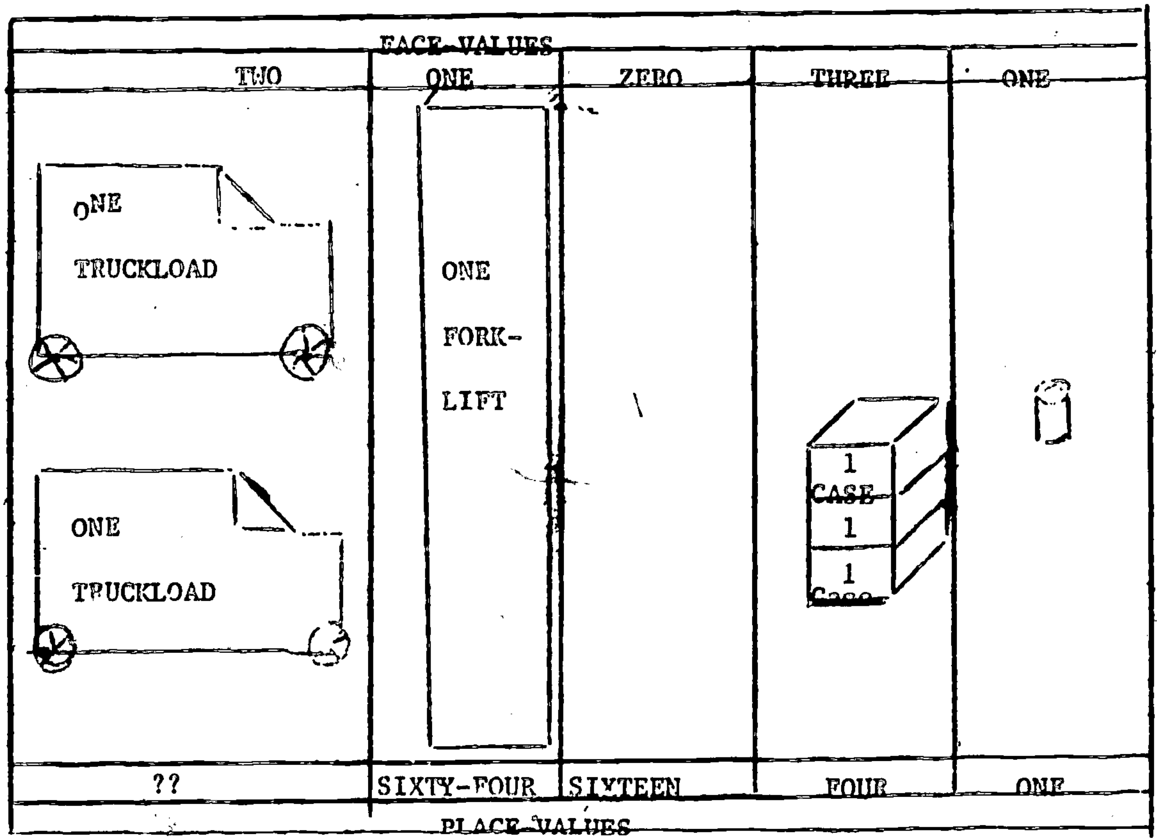
one tablespoon	= four fluidrams	
one roller	= four tablespoons	
one cup	= four rollers	= one pan or quarter wall
one quart	= four cups	= one wall
one gallon	= four quarts	= one room
one case	= four gallons	= one apartment
one car basket	= four cases	= one unit
one fork lift	= four carbaskets	= one building
one truckload	= four fork lifts	= one quarter-block
one shipment	= four truckloads	= one block
one warehouse	= four shipments	= one quarter-subdivision
four warehouses		= one subdivision
four subdivisions		= one section
four sections		= one season
four seasons		= one year
four years		= one presidential term
four presidential terms		= one quarter life time
four quarter life times		= one lifetime

We can use the Base Four numeration system to effectively indicate a specific quantity of wall paint as well as the price.

In the Base Four numeration system, numerals are written to indicate the numbers. The numeral 312 is written to indicate the number fifty four. Each symbol of a numeral has a face-value and a place-value. For example in the numeral 312, the face-value of 2 is two and the place value of 2 is one. The face-value of 1 is one and the place value of 1 is four. The face-value of 3 is three and the place-value of 3 is sixteen. The corresponding Base Ten numeral is given by the representation 3 x 16 + 1 x 4 + 2 x 1 or 54. Thus, the number that these numerals represent is fifty-four. In the figure below we have fifty four units of paint. What are these units?



Example 1. State the values shown by each symbol in the Base Four numeral 21031.



Answer The 2 is a symbol for two two-fifty-sixes.

The first 1 is a symbol for one sixty-four.

The 0 is a symbol for zero sixteens.

The 3 is a symbol for three fours.

The second 1 is a symbol for one one.

Hence, the corresponding Base Ten numeral is given by

2 x 256 + 1 x 64 + 0 x 16 + 3 x 4 + 1 x 1 or 589.

The corresponding Base Eight numeral is given by

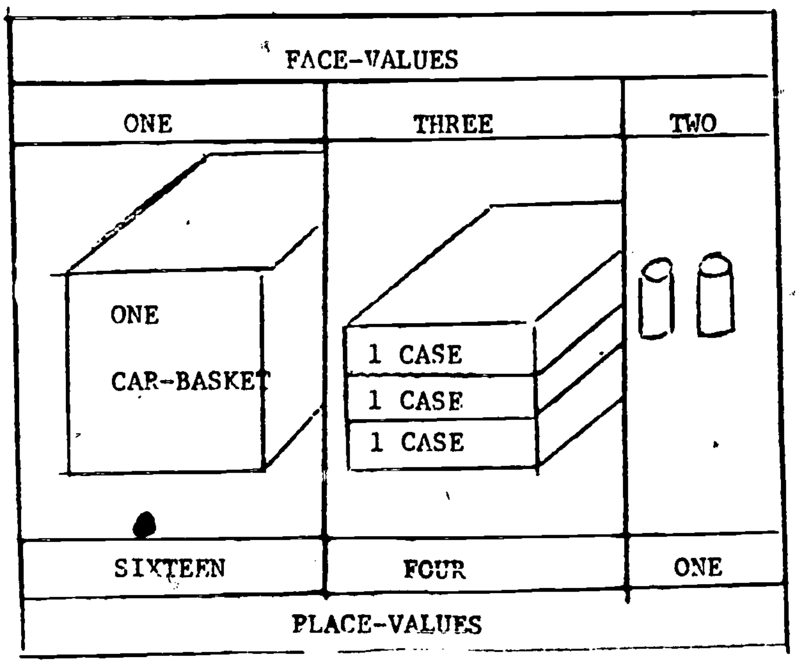
2 x 400 + 1 x 100 + 0 x 20 + 3 x 4 + 1 x 1 or

1000 + 100 + 0 + 14 + 1 or 1115.

Example 2. How many gallons of paint must be added to the paint in Example 1 in order to make a total of 100000 gallons of paint (Base Eight)?

Answer	We need	3	gallons to complete the case	3 gallons
	We need	0	additional cases to complete the carbasket of paint.	0 gallons
	We need	3	more carbaskets of paint to complete the forklift.	300 gallons
	We need	2	more forklifts of paint to	2000 gallons
	We need	1	more truckload of paint to complete the shipment	10000 gallons
	We need a total of			12303 gallons

Example 3. State the values shown by each symbol in the Base Four numeral 132.



Answer: The 1 is a symbol for one sixteen.

The 3 is a symbol for three fours.

The 2 is a symbol for two ones.

Give the corresponding Base Ten numeral.

Give the corresponding Base Eight numeral.

Games:

(a) Use the Paint Model to indicate the Base Four numeral 3.

(b) Use the Paint Model to indicate the Base Four numeral 30.

(c) Use the Paint Model to indicate the Base Four numeral 201.

(d) Use the Paint Model to indicate the Base Four numeral 21.

(e) Use the Paint Model to indicate the Base Four numeral 112.

(f) Use the Paint Model to perform the indicated operations:

$$\begin{array}{r} 2301 \\ +\ \underline{1011} \end{array} \qquad \begin{array}{r} 3233 \\ -\ \underline{1211} \end{array} \qquad \begin{array}{r} 3231 \\ -\ \underline{2133} \end{array}$$

Exercises:

1. Complete the following table.

Base Four	3	30	201			
Base Ten				9	70	77

* * *

Modify FORTRAN PROGRAM NO. 1 in order to write FORTRAN PROGRAM NO. 6 so that the answers to Problem 1 above may be computed.

*

2. The first four even whole numbers are represented by 0, 2, 10, and 12 in Base Four. List the first twenty even Base Four representations. How can you tell if a Base Four number is even?

* * *

Modify FORTRAN PROGRAM NO. 3 to write FORTRAN PROGRAM NO. 7 that will help to obtain the list requested in Problem 2.

*

3. List the first twenty multiples of four in Base Four. Tell how one would recognize a multiple of four representation in Base Four.

4. The first four odd whole numbers are represented in Base Four by 1, 3, 11, and 13. List the first twenty odd numbers by Base Four representations. How can you tell whether or not a numeral indicates a Base Four odd number?

5. Use the Paint Model to answer the following questions:

   (a) How many gallons are there in a warehouse of paint?

   (b) How many gallons are there in a car-basket of paint?

   (c) What is a subdivision of paint?

   (d) What is a lifetime of paint?

* * *

Modify FORTRAN PROGRAM NO. 5 to write FORTRAN PROGRAM NO. 8 to solve Problem 3 above.

*

2.5 Base Two As A System of Numeration*

In Base Two, we use the symbols 0 and 1. We use the numerals

0 to represent the number of full gallons in a quart,

1 to represent the number of full quarts in a liter,

10 to represent the number of pints in a quart,

11 to represent the number of angles in a triangle,

100 to represent the number of quarts in an American gallon,

101 to represent the number of quarts in a Canadian gallon,

111 to represent the number of days in a week, and

1000 to represent the number of pints in an American gallon.

Exercises

1. Using the Base Two Numeration System, give numerals to represent

   (a) the number of gills in a gallon.

   (b) the number of pints in a half an American gallon.

   (c) the number of weeks in the year.

   (d) the number of cups in a quart.

   (e) the number of quarts in a half-gallon.

   (f) the number of quarts in three gallons.

2.6 Place Value and Face Value In The Base Two Numeration System

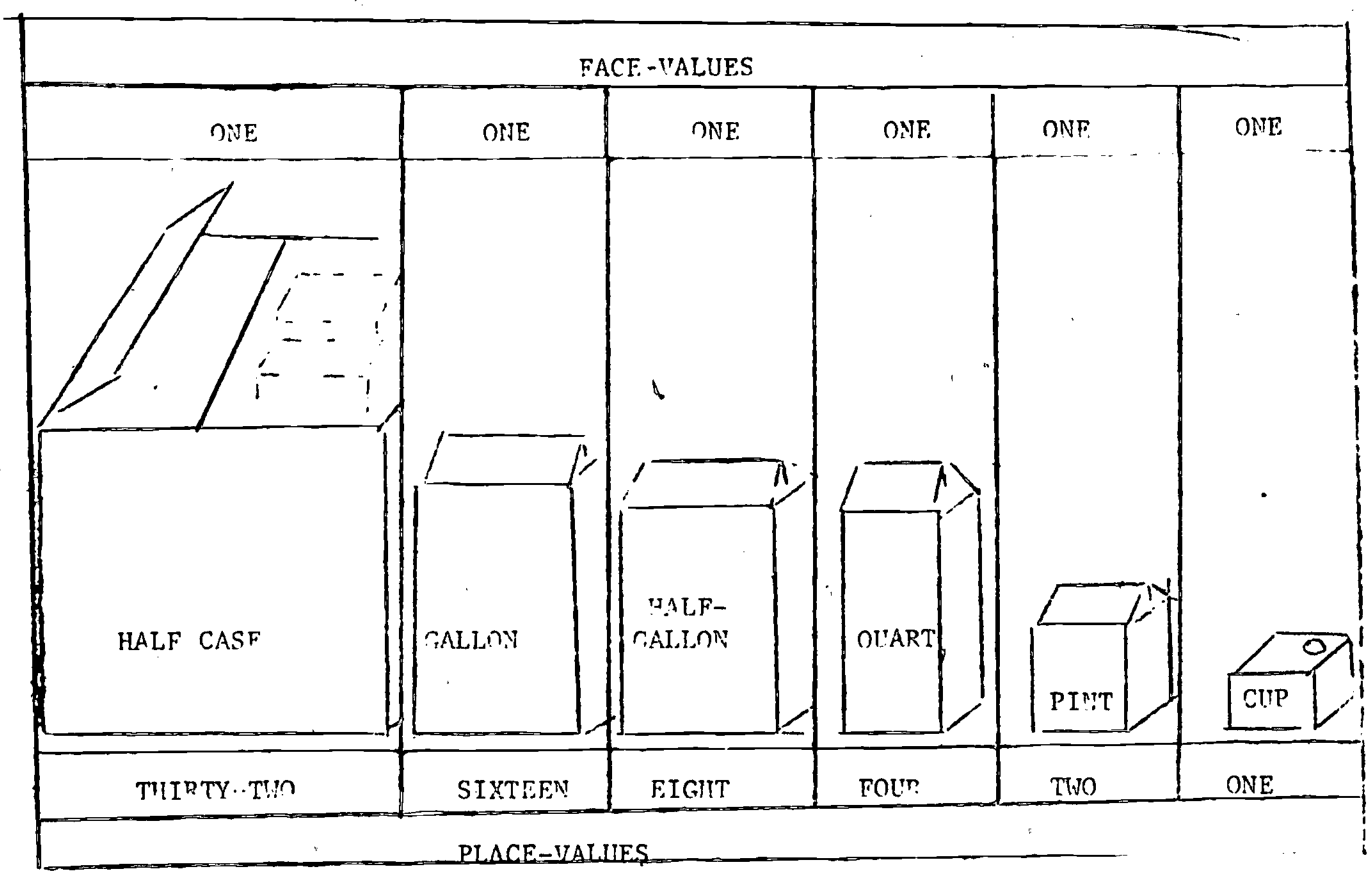
In the Base Two Numeration system numerals are written to indicate the numbers. The six place numeral 111111 is written to indicate the number sixty-three. Each symbol of a numeral has a face-value and a

*One way to write from 1 to 100 in Base Two is to write from 1 to 100 in Base Ten skipping any number having a numeral representation containing 2, 3, 4, 5, 6, 7, 8, or 9.

place-value. For example, in the numeral 111111, the face-value of the first 1 is 1 and the place-value of that 1 is thirty-two. The face-value of the second 1 is 1 and the place-value of that 1 is six-teen. The face-value of the third 1 is 1 and the place-value of that 1 is eight. The face-value of the fourth 1 is 1 and the place-value of that 1 is four.

<u>Question</u>: Can you give the face-values and place-values for the last two symbols in the Base Two numeral 111111? If so, what are they?

We can use a Base Two numeration system to effectively indicate the measure of a liquid. For example, 111111 cups represents one half case, one gallon, one half-gallon, one quart, one pint, and one cup.



Fixed and Floating Points

In dealing with the computations on a digital computer, it is necessary that the students are familiar with the two numeration systems- (1) Fixed points and (2) Floating point.

(1) A fixed point system is a numeration system in which the position of the point is fixed with respect to one end of the numeral. Thus, fixed points are integers.

In dealing with fixed point numbers, one must realize that the computer gives out only integer form of values as a result of computation. In other words, the computer truncates all the fractional part of the number.

For example: If 16 is divided by 3, then the answer given out by the computer will be 5, even though the correct answer should be 5.3333 -------. Thus the decimal portion of the number is truncated. Hence, one must be careful and specify if and when the computation and the result of the ocmputation are desired in integer form.

(2) Floating point numbers are the numbers in which the fixed position of the decimal point is variable in accordance with the requirement of the computation.

There are three different ways of representing the floating points. They are:

(a) Conventional floating form: Example: The numbers of the form 42.225, 3.2534 ...etc.

(b) Ordinary Scientific floating point form:

Example: $42.4225\ (10^0)$; 42.4225×10^0; 42.4225×10^{00};

0.424225×10^2; 0.004225×10^{04}; $.004225 \times 10^4$;

42422500×10^{-6} ... many other variations are possible.

(c) Exponential floating point form:

Example: 42.4225E00; 42.4225E + 00; 4.24225E01;

4.24225E1; .424225E02 ... Etc., ... Etc.

(Further discussion on these forms can be found in the discussion of the Format statements.)

Why the exponential floating point form?

To answer this question, consider the following example of multiplication.

```
      n n n . d d d d
        m m .   d d d
  k k k k k . d d d d d d d
```

This shows that the decimal point in the product is located so that it is to the left of the sum of the decimal places in both the multiplier and multiplicant.

Furthermore; it is possible to set the number of decimal places in advance whether we are using hand computation or a digital computer. Suppose we decide to have our computation to three decimal places. The disadvantage of this system will be formed in the following illustration:

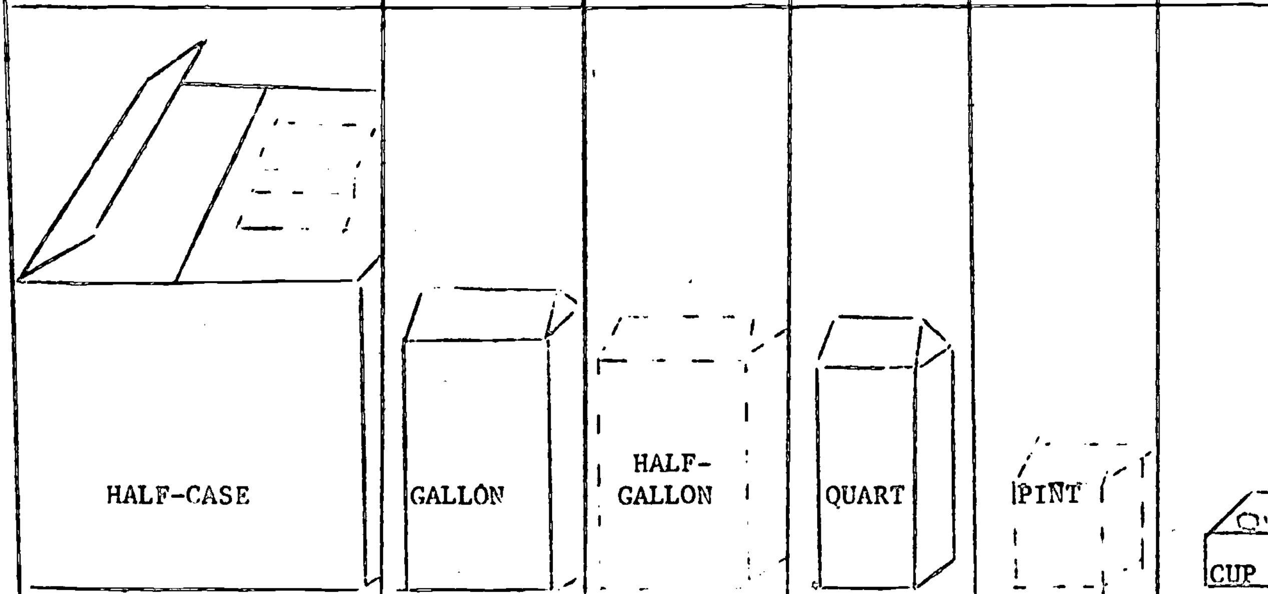
$$\begin{array}{r} 0.058 \\ \times\ \ 0.025 \\ \hline 0.000145 \end{array}$$

Since we are limited to the three digits, we lose all the non-zero digits.

To avoid this kind of difficulties, most computers engaged in scientific computation use the floating point arithmetic. In computers certain circuits are built in so that the decimal points will be aligned so that exponents are equal before addition and subtraction. In multiplication exponents are added and in division the exponent of the divisor is subtracted from the exponent of the divid'nd. All these are done automatically without requiring any special program instruction. Thus the computation is fast and efficient. Hence it is the advantage of using the third form.

Example 1. State the place-value shown by each symbol in 110101 where the numeral is given in Base Two. Also state the corresponding face-values.

FACE-VALUES					
ONE	ONE	ZERO	ONE	ZERO	ONE



THIRTY-TWO	SIXTEEN	EIGHT	FOUR	TWO	ONE
PLACE-VALUES					

Example 2. How many cups of milk are needed to add to 1001 cups to get a gallon of milk?

Answer:

1 cup is needed to go with the one cup to get one pint	1 cup
1 pint is needed to go with this pint	10 cups
1 quart is needed to go with the quart thus obtained	100 cups
	111 cups

Games:

(a) Use the Milk Model to indicate the Base Two numeral 1.

(b) Use the Milk Model to indicate the Base Two numeral 101.

(c) Use the Milk Model to perform the indicated operations:

$$\begin{array}{r} 101 \\ +\underline{100} \end{array} \qquad \begin{array}{r} 111 \\ -\underline{011} \end{array} \qquad \begin{array}{r} 10110 \\ -\underline{101} \end{array} \qquad \begin{array}{r} 1111 \\ +\underline{1111} \end{array}$$

Exercises:

(1) Complete the following table.

Base Two	1	10	110				
Base Ten				212	45	128	16

(2) Expand the Base Two numeral 10111 in Base Ten.

(3) List the first sixteen even numbers given in Base Two.

(4) List the first thirty positive integers given in Base Two.

(5) Tell which of the following Base Two numbers are multiples of four.

(a) 11011 (d) 10110

(b) 110000 (e) 1000010

(c) 11111 (f) 1010000

(6) Tell which of the following Base Two numbers are each integral powers on 10:

(a) 1100 (c) 11000

(b) 10000000000 (d) 1000

2.7 Comparison of the Base Two System and the Base Eight System With the Base Four System

Example 1a. 000_{two} represents the same number as 0_{eight}

Example 1b. 001_{two} represents the same number as 1_{eight}

Example 2a. 010_{two} represents the same number as 2_{eight}

Example 2b. 011_{two} represents the same number as 3_{eight}

Example 3a. 100_{two} represents the same number as 4_{eight}

Example 3b. 101_{two} represents the same number as 5_{eight}

Example 3c. 110_{two} represents the same number as 6_{eight}

Example 3d. 111_{two} represents the same number as 7_{eight}

Example 4a. $000\ 000_{two}$ represents the same number as 00_{eight}

Example 4b. $001\ 000_{two}$ represents the same number as 10_{eight}

Example 5a. $010\ 000_{two}$ represents the same number as 20_{eight}

Example 5b. $011\ 000_{two}$ represents the same number as 30_{eight}

Example 6a. $100\ 000_{two}$ represents the same number as 40_{eight}

Example 6b. $101\ 000_{two}$ represents the same number as 50_{eight}

Example 6c. $110\ 000_{two}$ represents the same number as 60_{eight}

Example 6d. $111\ 000_{two}$ represents the same number as 70_{eight}

Example 7a. $0\ 000\ 000_{two}$ represents the same number as 000_{eight}

Example 7b. $1\ 000\ 000_{two}$ represents the same number as 100_{eight}

Consequently,

$111,\ 111,\ 111_{two}$ represents the same number as $7\ 7\ 7_{eight}$,

$101,\ 100,\ 111_{two}$ represents the same number as $5\ 4\ 7_{eight}$,

and $100,\ 010,\ 001_{two}$ represents the same number as $4\ 2\ 1_{eight}$.

We easily determine that

$4\ 5\ 6_{eight}$ represents the same number as $100,\ 101,\ 110_{two}$

and $4\ 5\ 1_{eight}$ represents the same number as $101,\ 001,\ 001_{two}$. 4 or 5 examples for Base 4 to Base 2.

Since it is relatively easy to convert from Base Two to Base Eight, from Base Eight to Base Two, from Base Two to Base Four and from Base Four to Base Two, it should be almost as easy to convert from Base Four to Base Eight or from Base Eight to Base Four. We need only convert from the first base to Base Two and then convert to the second base. For example,

$1\ 3\ 1\ 2_{four} = 1\ 11\ 01\ 10_{two} = 1\ 110\ 110_{two} = 1\ 6\ 6_{eight}$

$1\ 7\ 3_{eight} = 1\ 111\ 011_{two} = 1\ 11\ 10\ 11_{two} = 1\ 3\ 2\ 3_{four}$.

Exercises

1. Complete the following table.

Base Eight	1	1 1	1 5	3 7	
Base Two	001	001 001			101

2. Complete the following table.

Base Eight	7 0	4 1	3 5	3 1 4	
Base Two					111010

3. Complete the following table.

Base Two	101	1010	1101	10001	
Base Eight					26

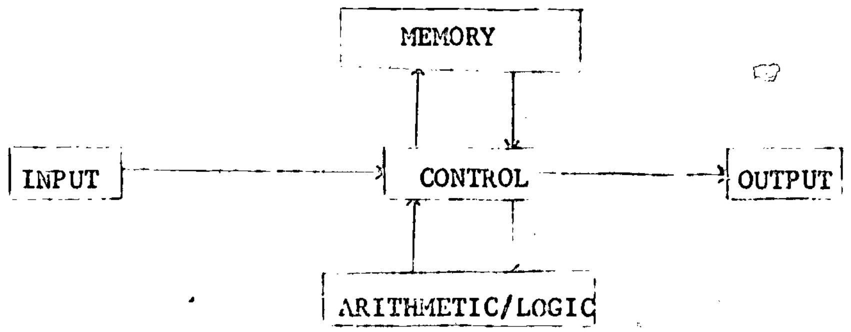
4. Write programs to:

(a) Change a number from Base 8 to Base 4.

(b) Change a number from Base 4 to Base 8.

Design of a Simple Computer

The relationship among the principal components of the computer can be understood by examining the figure below.



Input to the computer may be by punched cards, tape, teletype, or directly from the keyboard on the console. Input consists of the set of statements which make up the program. The statements may be instructions to the computer to perform certain operations or data to be used in the processing of the program.

The memory unit is where information is stored until needed in the execution of some part of the program. This is the central part of the ocmputer. All information must pass through it, including data for manipulation and instructions which tell the machine what to do.

The arithmetic/logic unit is where the actual computations take place. This unit may add, subtract, multiply or divide: higher-order mathematics is not done directly. This unit also performs certain logical operations; it can determine if a number is positive or negative, zero or non-zero.

The control unit acts in a supervisory capacity and exercises control over all units in the system. It coordinates the activities of the other units by timing and directing the flow of information from one unit to another.

The output section records in convenient form the contents of memory (answers to problems, original data, etc.). Output may be in the form of punch cards, punched tape, magnetic tape or printed sheets.

Let us design a simple computer and see how it may be operated to solve some simple problems.

First of all we will design a small memory which consists of 25 words, each one has a capacity of storing a two digit number and sign plus or minus.

C	E	T	H	D
O	K	V	Z	W
I	S	Q	Y	J
R	M	P	U	X
A	F	L	G	B

Memory of Simple Computer

Questions :

1. What is the smallest number which can be put in the memory of our simple computer?
2. What is the largest number which can be put in the memory of our simple computer?
3. What sort of instructions should we give our computer to execute? Make a table of them and assign a number to each of them.

Sample Table

Operations	Instructions		Abbreviation	
Arithmetic	Clear and Add		CLA	1
	Add		ADD	2
	Subtract		SUB	3
	Multiply		MUL	4
INPUT & OUTPUT	Divide		DIV	5
	Stφre		STφ	6
	Read		READ	7
	Print		WRITE	8
CONTROL	Transfer	Unconditional	Gφ Tφ	9
		Conditional	IF THEN	0

4. How would you instruct our simple computer to add two numbers, say 35 and 17?

First instruction will be to read the numbers 35 and 17 from tape or punched card and store in memory cells A and B respectively, where A and B are address of two cells in the computer memory. Second instruction will be to clear the Arithmetic Unit to zero and add A. Third instruction will be to add B. Fourth instruction will be to print out the result which is stored in the Arithmetic Unit.

In short, we can write the above instructions as follows:

READ A, B

CLA

ADD B

PRINT (the sum of A, B)

5. From the instruction table, assign the number which corresponds to each of the instructions in the program for the sum of two numbers.

6. Write a program that will instruct the computer to subtract two numbers, say 35, 17.

7. Write a program that will instruct the simple computer to multiply two numbers, say 35, 17.

8. Write a program to divide 95 by 5 using the method of repeated subtraction.

Consider the following statements in the program

Read (1,11) A(1), A(2), A(3)

11 Format (3F 5.2)

(a) What variables are read by the machine?

(b) What does 11 refer to in the Read statement?

(c) What does F refer to in the format statement?

(d) Supposing A(1) = 5.6, A(3) = 132.75, A(2) = 9468.39. State how the machine would read these.

Exercise #2 Read (5,60) A,B,C,J,K 60

FORMAT (3 F 12.4, 2 I 3)

(a) How many variables are read in the above exercise?

(b) In the Format statement what does 3 F 12.4 refer to what variables?

(c) What type of variables are J, K?

(d) Write the first and last statement of the program in problem number 4 by using a format statement.

(e) At some stage after the computer has gone through all the mathematical steps it must print out our results. That is done by the word <u>write</u>. It's operation is similar to the input operation.

e.g. Write (5,60) A(1), A(2), A(3), A(4)

60 FORMAT (A I 2)

Explain the above two statements.

[Note: teacher should provide some examples if necessary.]

There are two types of transfer statements

1. The unconditional transfer-simply states the computer to go to a certain line number such as Go To 6 simply states go next to line number 6.
2. Conditional transfer- There are two kinds of conditional statements, one is called logical IF and the other is called arithmetic IF. In the logical the decision is based on a logical quantity being true or false and in a quantity being either less than zero, zero or greater than zero in value.

A logical transfer statement is of the form

IF (expression) statement number

IF expression is true go to statement number, if false go to next statement number in the program.

The (expression) can relate two values by using one of six relational operations.

Relational Operation	Mathematical Symbol	Definition
GT	>	greater than
GE	≥	greater than or equal to
EQ	=	equal to
NE	≠	not equal to
LE	≤	(__________)
LT	<	(__________)

The form of the arithmetic transfer is

IF (a) n_1, n_2, n_3

means if a < 0, Go To line number n_1

if a = 0, Go To line number n_2

if a > 0, Go To line number n_3

Describe what will happen under the conditions given below:

(a) IF (J. NE. 0) Go To 18

(b) IF (J-N) 5, 5, 20

(c) IF (IND) 10, 20, 30

```
10 X = Y
GO TO 40
20 X = 20*Y
GO TO 40
30 X = Y**2
40 STOP
```

(d) IF (7.GT. SUM) Go To 40

```
CAL = QUO883
Go To 30
40 CAL = QUO**Z
30 ........
SU = 7
QUO = 5.0
Z = 2
```

Do Loop

Have the students discuss how to read and store a list of 20 numbers in our computer so that the individual members of the list may be referred to. The object is to have them decide upon some number pattern for referring to the members of the list in the order given. The simplist, and hence the best, way is to number them from 1 to 20,e.g. A(1), A(2), ..., A(20).

The students may decide to read and store the members of the list all at once or to do it in groups of 10, 5, 4, 2 or 1. If one should suggest that the computer be told to repeat the read process, try to bring out the concept of a loop. However, do not force it. If it is brought out in this discussion, the next part will be good to reinforce the concept.

Then consider how to instruct the computer to calculate the sum of the sequence of the members of the list.* Note that only one number may be added at a time. Somewhere during this discussion the idea of having the computer repeat a process should come up. You should be able to generate a program somewhat like the following.

*Must allow the memory to store number with more than two digits and at least for the location where X is stored.

```
10 X = 0
12 I = 0
14 T = 1
16 CLA I
20 ADD T
22 STO I
24 CLA A(I)
28 MUL A(I)
30 ADD X
32 STO X
34 IF I < 20 THEN 16
36 PRINT X
```

(Assumes that the members of the list have been real and stored in the location labelled A(1), A(2), ..., A(20))

Then define an instruction Do A I = S, E to replace steps 12, 14, 16, 18, 20, 22 and 34. A represents the line number of the final step of the process which is to be repeated, I represents the name of the index which will count the number of times the process has been done, < represents the first value for I (generally 1) and E represents the last value for I. Hence, E-S would be the number of times that the process will be repeated.

The above program then becomes:

```
10 X = 0
20 DO 32 I = 1, 20
24 CLA A(I)
28 MUL A(I)
30 ADD X
32 STO X
36 PRINT X
```

Note that the values of S and E were determined here by the way in which we named the members of our list. The freedom of choice for S and E is to allow the index I to be used as we have used it in this program.

Here is an alternate way of writing the program which does not involve using the index during the repeated process-except as a count to determine when to stop. In this case the reading of the list into the computer is done simultaneously with the calculations.

```
100 X = 0
110 DO 160 I = 1, 20
130 READ A
130 CLA A
140 MUL A
150 ADD X
160 STO X
170 PRINT X
```

Have the students do some exercises such as the following:

1. Write a program to compute 10!

2. Write a program to calculate the sum of all multiples of 5 from 5 to 100.

3. Given a list of ten numbers, write a program to arrange these numbers in ascending order.

Mathematical Operations and Symbols

Symbols	Operations
+	Addition
-	Subtraction
*	Multiplication
/	Division
**	Exponentiation

Write the following examples in Fortran.

Example 1.

$Z = A(3B + C)^3$

Solution:

Z = A*(3*B + C) ** 3

The order in which the above operation will take place are as follows:

1) Multiply 3*B

2) Add the product of 3*B to C

3) Raise the sum of 3*B+C to the third power

4) Multiply the results by A.

Example 2.

$$y = \frac{-B + \sqrt{B^2 - 4AC}}{2A}$$

Solution:

```
Y = (-B+(B**2 - 4.0*A*C)**.5)/2.0*A
```

The order in which the above operations will take place are as follows:

1) Raise B to the second power
2) Multiply 4.0*A*C
3) Subtract B**2-4.0*A*C
4) Raise the remainder to the .5 power
5) Add -B to the results
6) Multiply 2.0*A
7) Divide the product into the results in 5).

Problems

Write the following algebraic expressions in the Fortran language.

1) $Y = AB + C$

2) $Y = A^2 + 2AB + B^2$

3) $Z = \frac{A(3B + 4C)^2}{3BC + D}$

4) $Y = \left[7\left(\frac{4A - 2AC}{3C}\right)^4\right]^{1/2}$

5) $Z = \left[\frac{(4X + 3Y)^3}{2XY}\right]^{1/2}$

6) $Y = 4\ (8 + \frac{5}{2^2})$

7) $Y = 4 \times 8 \div 2^4$