MATH BLOCK OPERATIONS

KEY	RESULTS/COMMENTS
A (TO POLAR) B (TO RECTANGULAR)	COORDINATE CONVERSION: POLAR COORDINATES: $\theta \rightarrow Y$, $R \rightarrow X$ RECTANGULAR COORDINATES: $y \rightarrow Y$, $x \rightarrow X$
C (RECALL)	$b \rightarrow Y$ $a \rightarrow X$
D (ACCUMULATE −)	$b - Y \rightarrow b$ $a - X \rightarrow a$
E (ACCUMULATE +)	$b + Y \rightarrow b$ $a + X \rightarrow a$
F (DEFINABLE $f()$)	EXECUTES SPECIAL SUBROUTINE, SEE PAGE 2-21*.
G ($\|y\|$)	$\|Y\| \rightarrow Y$
H (x^y)	$X^Y \rightarrow X$
I ($\ln x$)	NATURAL LOG (X) → X
J (e^x)	(2.718..RAISED TO CONTENTS OF X) → X
L (arc)	USE TO PREFIX ANY KEY BELOW IN ORDER TO COMPUTE RESPECTIVE INVERSE CIRCULAR FUNCTION.
M ($\sin x$)	$\sin X \rightarrow X$
N ($\cos x$)	$\cos X \rightarrow X$
O ($\tan x$)	$\tan X \rightarrow X$

KEY(S)	RESULTS/COMMENTS
K (TABLE (N))	PREFIXES THE FOLLOWING OPERATIONS (NOT AVAILABLE WITH A SINGLE KEY).
K (TABLE (N)) 1	SETS DEGREES
K (TABLE (N)) 2	SETS RADIANS
K (TABLE (N)) 3	SETS GRADS
K (TABLE (N)) 4	COMMON LOG(X) → X
K (TABLE (N)) 5	$10^x \rightarrow X$
	ANGLE CONVERSION:
K (TABLE (N)) 6	D,M,S → DECIMAL DEGREES
K (TABLE (N)) 7	DECIMAL DEGREES → D,M,S
	(6, 7: D → Z, M → Y, S → X; Dec. Deg. → X)
K (TABLE (N)) 8	X FACTORIAL → X $0 \leqslant \text{int} \|X\| \leqslant 69$
K (TABLE (N)) 9	(Y ROUNDED TO POWER OF 10 INDICATED BY CONTENTS OF X) → X
K (TABLE (N)) FMT	PLOTTER SCALING FUNCTION, SEE PAGE 2-15*.
K (TABLE (N)) CLEAR x	CLEARS ALL NUMERIC REGISTERS’
K (TABLE (N)) SUB RETURN	‘DO-LOOP’ SUBROUTINE, SEE PAGE 2-22*.

*For further information on any of these operations, see the Math Block Operating Manual.

SPECIFYING TRIGONOMETRIC UNITS

The units to be used in problems involving trigonometric function or vector arithmetic are easily selected according to the procedure listed in Table 2. The units specified, either degrees, radians, or grads, are indicated by the appropriate light above the Math key-block.

Table 2. Specifying Units

TO SPECIFY:	PRESS:	INDICATION: (1) degrees	(2) radians	(3) grads
DEGREES	K / TABLE (N) 1	●	○	○
RADIANS	K / TABLE (N) 2	○	●	○
GRADS	K / TABLE (N) 3	○	○	●

> **NOTE**
>
> If units are not specified after the calculator is turned on, the calculator will automatically select degrees when the first trigonometric or coordinate conversion key is executed.

TRIGONOMETRIC FUNCTIONS

The circular functions of angles having an absolute magnitude less than 10^{11} may be computed by using the keys listed below. When evaluting angles greater than the above limit, refer to the APPENDIX for a summation of calculator error. The inverse circular functions are computed within the range of principal values of each function:

$$-90^\circ \leq \mathrm{SIN}^{-1}\ X \leq 90^\circ$$
$$0^\circ \leq \mathrm{COS}^{-1}\ X \leq 180^\circ$$
$$-90^\circ \leq \mathrm{TAN}^{-1}\ X \leq 90^\circ$$

For instance: $\cos 150^\circ = \cos 210^\circ = \cos 510^\circ = (\text{etc.}) = -.866$

But: $\cos^{-1} -.866 = 150^\circ$

TRIGONOMETRIC FUNCTIONS

> **NOTE**
> The STATUS light will indicate an attempted calculation which is beyond the above range of inverse circular functions.

SINE [M / sin x]

[M / sin x] Calculates the Sine of the X-register contents and inserts the result in X.

COSINE [N / cos x]

[N / cos x] Calculates the Cosine of the X-register contents and inserts the result in X.

TANGENT [O / tan x]

[O / tan x] Calculates the Tangent of the X-register contents and inserts the result in X.

INVERSE FUNCTIONS [L / arc]

[L / arc] When followed by a trigonometric key, will calculate the respective inverse circular function of the X-register contents and insert the result in the X-register.

EXAMPLE:

Calculate $\sin^{-1} 0.5$

PRESS: [FIX( )] [4]

PRESS: [K / TABLE (N)] [1] (Set degrees)

PRESS: [•] [5] [L / arc] [M / sin x]

DISPLAY: 30.0000 → x (Degrees)

To calculate the sine 30°:

PRESS: [M / sin x]

DISPLAY: .5000 → x

LOGARITHMIC AND EXPONENTIAL FUNCTIONS

The following keys add logarithmic and additional exponential functions to your Model 10.

COMMON LOGARITHM

[K / TABLE (N)] [4]

[K / TABLE (N)] [4] Calculates the logarithm, to base 10 (common logarithm), of the X-register contents and displays the result in X.

ANTILOG, BASE 10

[K / TABLE (N)] [5]

[K / TABLE (N)] [5] Raises 10 to the power indicated by the X-register contents and displays the result in X (i.e. 10^x or $\log_{10}^{-1} x$).

EXAMPLE:

Raise 10 to the power 0.69897

PRESS: [FIX ( )] [7]

PRESS: [•] [6] [9] [8] [9] [7]

PRESS: [K / TABLE (N)] [5]

DISPLAY: 5.0000000 → x

To take the $\log_{10}$ of 5

PRESS: [K / TABLE (N)] [4]

DISPLAY: .6989700 → x

LOGARITHMIC AND EXPONENTIAL FUNCTIONS

NATURAL LOGARITHM

[I / ln x] Calculates the logarithm, to base e (natural logarithm), of the X-register contents and displays the result in X.

ANTILOG, BASE e

[J / e^x] Raises e (2.718....) to the power indicated by the X-register contents and displays the result in X. (i.e. $\ln^{-1}x$).

EXAMPLE:

$\sqrt[5]{32}$

PRESS: [FLOAT]

PRESS: [3] [2] [ln x] [↑]

PRESS: [5] [÷] [↓] [e^x]

DISPLAY: 2.0000000 00 → x

[H / x^y] Raises the X-register contents to the power indicated by the Y-register contents. The result is displayed in the X-register, and the Y-register contents remain unchanged.

EXAMPLE:

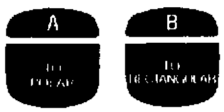
Find $19^{1.6}$

PRESS: [FLOAT] [1] [•] [6] [↑]

PRESS: [1] [9] [x^y]

DISPLAY: 1.111746476 02 → x

COORDINATE CONVERSION



VECTOR ARITHMETIC

The vector keys provide capability for performing complex and vector arithmetic with single keystroke operation.

Converts rectangular coordinates, consisting of x and y components in the X- and Y-registers respectively, to polar coordinates. The final display is:

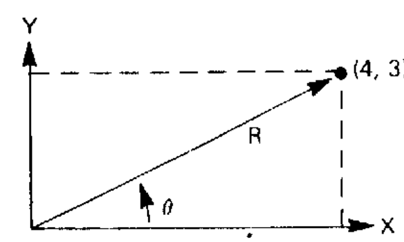
temporary z		
accumulator y	(angle θ)	$\theta = \tan^{-1}\frac{y}{x}$
keyboard x	(radius R)	$R = \sqrt{x^2 + y^2}$

When converting from rectangular (cartesian) to polar coordinates, the calculated angle θ will be within the range:

$$-180^\circ \leqslant \theta \leqslant 180^\circ$$

EXAMPLE:

Convert the coordinates 4, 3 (x, y) to polar degree form.



PRESS: [K TABLE (N)] [1] (Set Degrees)

PRESS: [FIX ( )] [3]

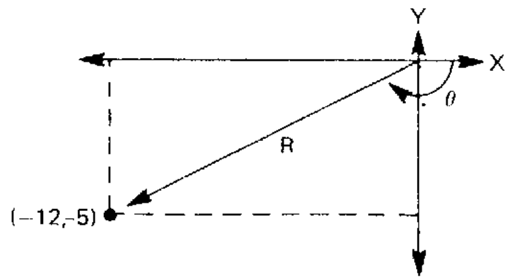
PRESS: [3] [↑] [4] [A]

DISPLAY: 36.870 → y (θ)
5.000 → x (R)

VECTOR ARITHMETIC

EXAMPLE:

Convert the coordinates –12, –5 (x, y) to polar radian form.



PRESS: [K / TABLE (N)] [2] [FIX ( )] [3] (Set radians)

PRESS: [CHG SIGN] [5] [↑] [CHG SIGN] [1] [2]

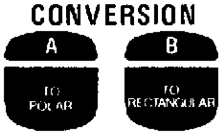
PRESS: [A / TO POLAR]

DISPLAY: -2.747 → y (θ, in radians)
13.000 → x (R)

Converts polar coordinates, when the radius (R) and the angle (θ) are in the X- and Y-registers respectively, to rectangular coordinates. The final display is:

temporary z		
accumulator y	(Y component)	Y = R sin θ
keyboard x	(X component)	X = R cos θ

COORDINATE CONVERSION

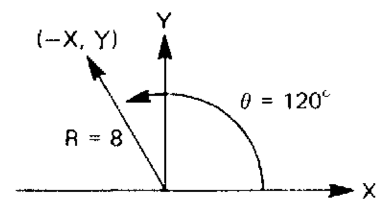


(continued)

VECTOR ARITHMETIC

EXAMPLE:

R = 8, θ = 120° (or −240°): convert to rectangular form.



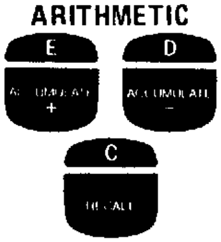
PRESS: [FIX ( )] [3] [TABLE (N)] [1]

PRESS: [1] [2] [0] [↑] [8]

PRESS: [B TO RECTANGULAR]

DISPLAY: 6.928 → *y* (Y component)
-4.000 → *x* (X component)

VECTOR ARITHMETIC



ACCUMULATE +, ACCUMULATE −, and RECALL are storage and recall keys which are associated with the *a* and *b* data storage registers. These keys provide complete capabilities for vector addition and subtraction.

> **NOTE**
>
> The *a* and *b* registers are cleared by the CLEAR key, thus CLEAR instructions should be used carefully when using these registers.

Adds (simultaneously) the contents of the X- and *a* -registers together and the contents of the Y- and *b* -registers together. The sums are entered in the *a* - and *b* -registers respectively, while the X- and Y-registers remain unchanged.

Thus:

$b + Y \rightarrow b$

$a + X \rightarrow a$

VECTOR ARITHMETIC

Subtracts (simultaneously) the contents of the X-register from the *a*-register contents and the contents of the Y-register from the *b*-register contents. The remainders are entered in *a*- and *b*-registers respectively, while the X- and Y-registers remain unchanged.

Thus:

$$b - Y \rightarrow b$$
$$a - X \rightarrow a$$

Recalls the contents of the *a*-register to the X-register and the contents of the *b*-register to the Y-register. The contents of the *a*- and *b*-registers remain unchanged.

Thus:

$$b \rightarrow Y$$
$$a \rightarrow X$$

The following examples demonstrate the use of the vector arithmetic keys.

EXAMPLE:

Vector addition

$(2x + 3y) + (4x + 5y) - (3x - 6y) = 3x + 14y$

PRESS: [FIX ( )] [3]

Clear *a*- and *b*-registers:

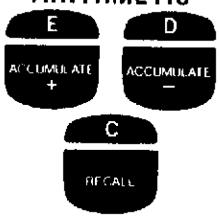
PRESS: [CLEAR]

or

PRESS:* [C RECALL] [D ACCUMULATE −]

*This operation clears the *a*- and *b*-registers but does not affect the FLAG or the Z-register contents.

VECTOR ARITHMETIC



(continued)

VECTOR ARITHMETIC

Enter the first term:

PRESS: [3] [↑] [2] [E ACCUMULATE +]

RESULT: 0 + 3 = 3 → *b*
0 + 2 = 2 → *a*

Enter the second term and add:

PRESS: [5] [↑] [4] [E ACCUMULATE +]

RESULT: 3 + 5 = 8 → *b*
2 + 4 = 6 → *a*

Enter the last term and subtract:

PRESS: [CHG SIGN] [6] [↑] [3] [D ACCUMULATE −]

RESULT: 8 − (−6) = 14 → *b*
6 − 3 = 3 → *a*

Display result in X- and Y-registers:

PRESS: [C RECALL]

DISPLAY: 14.000 → y
3.000 → x

EXAMPLE:

Multiplication of complex numbers

(3 + j4)(−2 + j3) = −18 + j1

where: $j = \sqrt{-1}$

This type of problem (containing complex terms) is best solved in the calculator by the following method:

1. convert the quantities in parentheses to polar form ($R \angle \theta$),
2. multiply the R quantities using logarithms and add the angles,
3. convert the result back to rectangular form.

VECTOR ARITHMETIC

Clear the *a* - and *b* -registers:

PRESS: [CLEAR]

Enter (3 + j4) and convert to the polar form (R_1 $\angle\theta_1$):

PRESS: [4] [↑] [3] [A / TO POLAR]

Take the $\log_e$ of R_1 and store ln R_1 in the *a* -register and $\angle\theta_1$ in the *b* -register:

PRESS: [I / ln x] [E / ACCUMULATE +]

Enter (−2 + j3) and convert to the polar form:

PRESS: [3] [↑] [CHG SIGN] [2] [A / TO POLAR]

Take the $\log_e$ of R_2 and add ln R_2 and $\angle\theta_2$ to *b* - and *a* -registers respectively:

PRESS: [I / ln x] [E / ACCUMULATE +]

Take the $\log^{-1}$ (ln R_1 + ln R_2):

PRESS: [C / RECALL] [J / e^x]

Convert the result (R in X and $\angle\theta$ in Y) back to rectangular form:

PRESS: [B / TO RECTANGULAR]

DISPLAY: 1.000 → y (j)
-18.000 → x

The preceding procedure may be changed to division of complex numbers (i.e. [3 + j4] / [−2 + j3]) by changing the second ACCUMULATE + instruction to an ACCUMULATE − instruction.

SPECIAL FUNCTIONS

The TABLE key is also used in performing the following mathematical operations: angle conversion, plotter scaling, rounding a number to a specified power of ten, and calculating X factorial (X!).

ANGLE CONVERSION

[K / TABLE (N)] [6]

[K / TABLE (N)] [7]

[K / TABLE (N)] [6] Converts an angle expressed in degrees, minutes, and seconds to decimal degrees. The angle must be entered into the calculator as follows:

temporary z	D	(degrees)
accumulator y	M	(minutes)
keyboard x	S	(seconds)

The result in decimal degrees appear in X; the Y- and Z-registers are cleared.

[K / TABLE (N)] [7] Converts an angle expressed in decimal degrees to degrees, minutes, and seconds. The angle to be converted must be entered into the X-register, and the resulting angle appears as in the previous display.

EXAMPLE:

Angle conversion.

45° 30′ 3″ = 45.50083°

PRESS: [FIX( )] [5]

PRESS: [4] [5] [↑] [3] [0] [↑]

PRESS: [3] [K / TABLE (N)] [6]

DISPLAY: 45.50083 → x (decimal angle)

To convert back to degrees, minutes, seconds:

PRESS: [K / TABLE (N)] [7]

DISPLAY: 45.00000 → z (D)
30.00000 → y (M)
3.00000 → x (S)

SPECIAL FUNCTIONS

When converting angles, the inverse conversion may not always recover the exact original angle. This is due to the round-off error inherent in the calculator's 12-digit numeric capability.

To observe this error:

PRESS: [FLOAT]

DISPLAY: 4.500000000 01 → z
3.000000000 01 → y
2.999999880 00 → x

ROUNDING A NUMBER

[K TABLE (N)] [9] Will round the number in the Y-register to the power of ten indicated by the integer value of the number in the X-register. The rounded number appears in X, while the Y-register remains unchanged.

EXAMPLE:

Round 5610.0 to 10^2 (or nearest 100).

PRESS: [FIX ( )] [3]

PRESS: [5] [6] [1] [0] [↑]

To enter a power of ten (10^2) and round the number:

PRESS: [2] [K TABLE (N)] [9]

DISPLAY: 5610.... → y
5600.000 → x (Y rounded)

To round the number in Y to another power of ten (10^4):

PRESS: [4] [K TABLE (N)] [9]

DISPLAY: 5610.000 → y
10000.000 → x (Y rounded)

SPECIAL FUNCTIONS

ROUNDING A NUMBER
[K / TABLE (N)] [9]
(continued)

A fractional number may be rounded by entering a negative number into the X-register.

EXAMPLE:

Round 0.025 to the 10^{-2} or nearest $\frac{1}{100}$.

PRESS: [•] [0] [2] [5] [↑]

PRESS: [CHG SIGN] [2] [K / TABLE (N)] [9]

DISPLAY: 0.025 → y
0.030 → x (Y rounded)

CALCULATE X FACTORIAL
[K / TABLE (N)] [8]

[K / TABLE (N)] [8] Will replace the X register contents with X!.

Where: $0 \leqslant \text{int } |X| \leqslant 69$.

EXAMPLE:

Calculate 7!

PRESS: [FIX ( )] [1] [7] [K / TABLE (N)] [8]

DISPLAY: 5040.0 → x

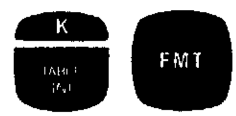
ABSOLUTE VALUE OF Y
[G / |Y|]

[G / |Y|] Sets the Y-register contents positive without affecting the sign of the exponent; the X- and Z-registers remain unchanged.

SPECIAL FUNCTIONS

PLOTTER SCALING

A problem usually encountered when writing a calculator/plotter program* is that of scaling the available X and Y problem-variables to X_{plot} and Y_{plot} coordinates which the plotter can use. The plotter scaling function to be described, simplifies this typical plotting problem.



Replaces X and Y plotter problem-variables which are entered in corresponding X- and Y-registers, with computed X_{plot} and Y_{plot} coordinates. X_{plot} and Y_{plot} coordinates are each simultaneously calculated by the equations:

$$X_{plot} = \frac{9999}{X_{max} - X_{min}} [X_{var} - X_{min}]$$

$$Y_{plot} = \frac{9999}{Y_{max} - Y_{min}} [Y_{var} - Y_{min}]$$

Where: X_{var} and Y_{var} = entered problem-variables
X_{max}, X_{min} = problem variable range
Y_{max}, Y_{min} = problem variable range
9999 = maximum unit range of X-Y plotter

The problem-variable range must be predetermined and entered into data registers 001 to 004 according to the following table:

Table 3. Variable Range Storage

PROBLEM-VARIABLE RANGE:	STORE IN REGISTER:
X_{min} →	001
X_{max} →	002
Y_{min} →	003
Y_{max} →	004

The stored variable range limits are automatically assumed as being the graph limits. Thus, by setting the lower left and upper right graph limit controls on the calculator plotter, the graph is calibrated to any required size and grid dimension (units per inch on graph).

*A plotting problem involves the following basic steps:

1. determine the problem-variable ranges.
2. determine the graph calibration (i.e. problem-units per inch on graph).
3. convert problem-variables to X_{plot} and Y_{plot} coordinates for graphing.
4. Write a program to calculate and plot coordinates.

PROGRAMMING FEATURES

This section describes the special program-related functions which the Math Block adds to your Model 10.

DEFINABLE FUNCTION

Is used to label and 'call' an often used function which is stored in your Model 10. The definable function may be executed at any time from the keyboard by pressing the DEFINABLE key, or it may be 'called' in a program by inserting the DEFINABLE (key) instruction.

The definable function is programmed similarly to a 'LABEL' subroutine; it is executed as a normal subroutine, except that it may be 'called' by a single keystroke or program step.

> **NOTE**
> The definable function should be stored near the end of the program memory to avoid accidental erasure from subsequent programming operations.

The following subroutine* is written as a definable function. Notice that the function must begin with LBL, DEFINABLE and its last step must be SUB/RETURN.

```
0490--LBL---51
0491-- F ---16
0492-- J ---75
0493-- UP---27
0494--1/X---17
0495-- + ---33
0496-- 2 ---02
0497--DIV---35
0498--S/R---77
0499-- 0 ---00
```

- 0490–0491: Definable Function address
- 0492–0497: Keys of function; may be any number of steps.
- 0498: Terminates function.

Figure 5. Programmed Definable Function

If the function were stored in your Model 10, you could use it anytime by pressing the DEFINABLE key.

*This subroutine computes the hyperbolic cosine of the X-register contents (cosh X) and places the result in the Y-register.

PROGRAMMING FEATURES

CLEARING STORAGE REGISTERS

will clear all numerical storage registers without affecting the *a* and *b* registers or the X, Y, and Z registers or the flag.

ITERATIVE SUBROUTINES (DO-LOOPS)

A programmed subroutine may be repeated n number of times by inserting the following keys at the end of the subroutine:

Where: (r) may be any key from 0 to 9 corresponding to the data storage register (reg. 000 through 009) that contains n; and n is equal to the absolute, integer value of the contents of the register designated. If $n < 1$ the subroutine will execute at least one time.

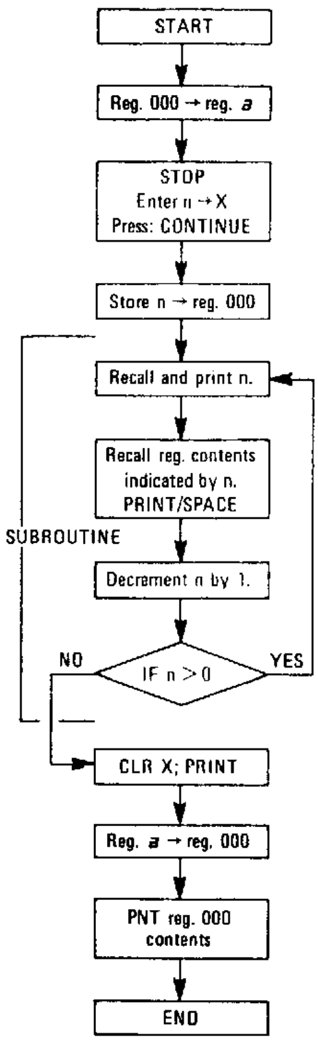
After the subroutine has been repeated 'n' times, the program exits from the subroutine* and resumes normal program execution at the step following the subroutine 'call instructions'. The branching instructions which 'call' the iterative subroutine *must* contain 6 program steps; if less than 6 steps are used, the program will probably not execute properly, but it will return to the correct step after exiting the subroutine.

The following program demonstrates the use of an iterative subroutine. The program is a register scanner, since it prints (or displays) the address and the contents of all data registers from the register number entered by the operator at the beginning of the program to register 000.

This program is written for use with a Model 10 containing a printer (Option 004). If your Model 10 does not have a printer, the note on Page 2-25 describes how to modify this program for the basic Model 10.

*The subroutine may be exited anytime before being repeating 'n' times by clearing the register containing 'n'.

PROGRAMMING FEATURES



Shaded steps are operations of iterative subroutine command.

STEP	KEY	CODE	
0000	CLR	20	
0001	XFR	67	
0002	0	00	Reg. 000 contents to reg. a.
0003	XTO	23	
0004	a	13	
0005	STP	41	Enter n. Press: CONTINUE.
0006	XTO	23	
0007	0	00	n stored in reg. 000.
0008	UP	27	
0009	GTO	44	
0010	S/R	77	
0011	0	00	'Call' subroutine.
0012	0	00	(Must be 6 steps.)
0013	1	01	
0014	8	10	
0015	GTO	44	
0016	3	03	
0017	0	00	
0018	XFR	67	
0019	0	00	
0020	PNT	45	
0021	UP	27	
0022	XFR	67	Subroutine:
0023	IND	31	Print n and print contents of reg. indicated by n.
0024	0	00	
0025	PNT	45	
0026	PNT	45	
0027	K	55	
0028	S/R	77	Iterative subroutine command.
0029	0	00	
0030	CLX	37	
0031	PNT	45	Print '0'.
0032	a	13	
0033	XTO	23	Reg. a contents to reg. 000.
0034	0	00	
0035	PNT	45	Print reg. 000 contents.
0036	PNT	45	
0037	PNT	45	
0038	PNT	45	
0039	PNT	45	
0040	PNT	45	Space printout above printer window.
0041	PNT	45	
0042	PNT	45	
0043	PNT	45	
0044	PNT	45	
0045	END	46	

Figure 6. Iterative Subroutine Program

After loading the program at step 0000:

PRESS: [RUN] [FLOAT] [END] [CONTINUE]