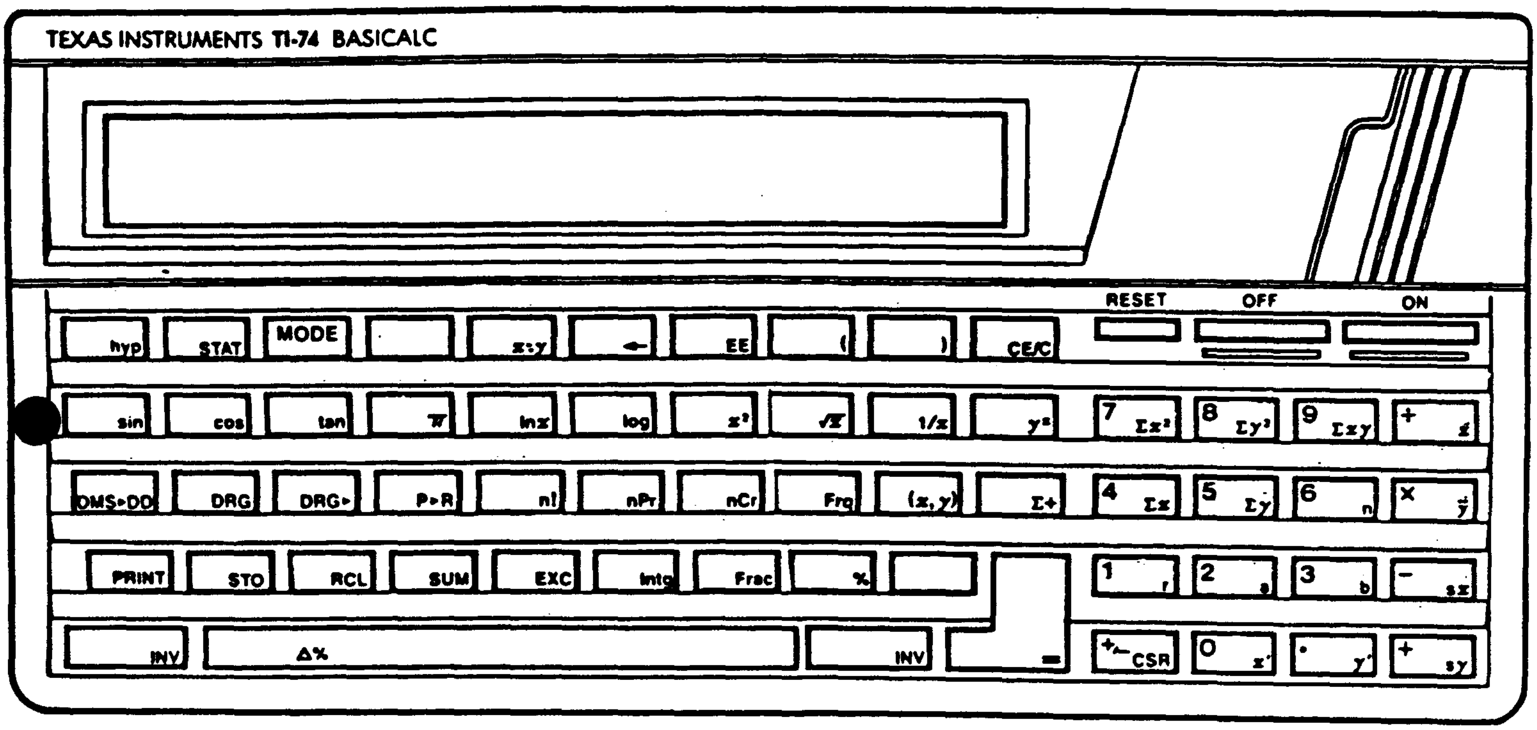
TI PPC NOTES

NEWSLETTER OF THE TI PROGRAMMABLE CALCULATOR CLUB
P.O. Box 1421, Largo, FL 34294

Volume 11, Number 2 Second Quarter 1986

The TI-74 arrives - see page 4



The full-size illustration above was extracted from pages ii and iii of the TI-74 User's Guide. The key nomenclature for the calculator mode is shown. Different key assignments are used for the BASIC mode; many of those functions are discussed on pages 4 ff. Generally, the key assignments in BASIC are very similar to those on the CC-40. The alphabet portion of the BASIC key assignment is pseudo-QWERTY. Note that the keys are not staggered from row to row.

ERRATA:

More on the fx-7000G - On V10N4P11 in the first line there is an exponent of 2 missing. The line should read "... It is a deriviative of the $(\sqrt{2})^2$ test by Brian Hayes ...".

On the same page the test from the "Computer Recreations" column of the April 1984 issue of Scientific American is titled "1.0000001 to the 27th Power". The title should have been "1.0000001 Squared 27 times".

Fast Graphics Mode Polar Plotter - V11N1P19 defined four examples for demonstration of the polar plotting routine. On V11N1P21/22 I reported that I was unable to get the plots defined for those examples except for example 1. It seems that there were errors in the functions for examples 2 through 4 on V11N1P19. The correct functions are:

Example 2

280	43	RCL
281	07	07
282	65	×
283	03	3
284	95	=
285	38	SIN
286	94	+/-
287	32	X:T
288	02	2
289	09	9
290	04	4
291	61	GTO
292	01	01
293	02	02
294	43	RCL
295	07	07
296	38	SIN
297	32	X:T
298	07	7
299	01	1
300	61	GTO
301	01	01
302	09	09

Example 3

280	43	RCL
281	07	07
282	35	1/X
283	65	×
284	01	1
285	08	8
286	00	0
287	55	÷
288	89	π
289	95	=
290	32	X:T
291	07	7
292	01	1
293	61	GTO
294	01	01
295	02	02

Example 4

280	43	RCL
281	07	07
282	65	×
283	02	2
284	95	=
285	39	COS
286	65	×
287	43	RCL
288	07	07
289	39	COS
290	95	=
291	32	X:T
292	07	7
293	01	1
294	61	GTO
295	01	01
296	09	09

where in example 3 it is not clear why GTO 102 is required rather than GTO 109.

Factorials on the CC-40 - V8N5P18 noted that factorials could be found on the CC-40 with the Mathematics module by recognizing that N! = Gamma(N+1). The final line stated that the CC-40 could then find factorials up to 85!. That should have been factorials up to 84! = 3.31424E+126.

MAGNETIC CARD AVAILABILITY FOR THE TI-59: Member J. M. Gallego has magnetic cards for sale. He will sell a box of 40 blank magnetic cards with carrying case for eight dollars ($8.00) each. Shipping is included. U.S. members should send money orders only to:

Q. Jose M. Gallego
250 Quintard Ave., Apt. 96
Chula Vista CA 92011/4924

TI-59 MATERIAL FOR SALE - V11N1P2 reported that former member William Vogel had donated his TI-59 equipment and documentation to the club, and offered it for sale at nominal prices. The remaining material includes:

Leisure Library Module and Documentation	$ 5.00
Three roll pack of old style printer paper	6.00
User Survival Guide (The Fish Book)	5.00

There are also a number of pads of the TI Program Record form. If you would like some of these send some postage and I will return as many pads as the postage allows.

A friend has made a defective TI-59 available to our members. The device has a curious malfunction. If you press 2nd-Op-16 after turn-on the display reads "160479.59". If you press 0-2nd-Op-17 the display reads "959." as expected, but another 2nd-Op-16 yields the display 160959.". If you go to LRN mode you can SST right on past step 959, but you cannot enter any code. In any partitioning you cannot access any of the data registers; however, arithmetic using the display register and the hierarchy registers is OK. The unit comes with a good battery, the Leisure Library module installed (but no acompanying documentation), a makeshift cover for the module, and a charger. There is some evidence of an unauthorized modification. There is no documentation. If you are interested, make an offer.

CC-40 MATERIAL FOR SALE - Maurice Swinnen has extra CC-40 material. He has divided the equipment into two systems, and will sell by system, not by individual components.

System 1:	CC40 with 18K Memory (new)	$ 375
	HX-1010 80 Column Thermal printer (new), with 1000 sheets of thermal paper, 500 sheets of glossy plain paper, 8 rolls of 8.5" x 100 ft thermal roll paper, and 6 boxes of thermal offset ribbons.	200
	HX-1000 Four Color Plotter-Printer (new)	200
	Mathematic Solid State Cartridge	60
	Electronics Engineering Solid State Cartridge	60
	Statistics Solid State Cartridge	60
	Mechatronic Disk Drive	295
	Ten Diskettes	50

	New Price	$1300
	Sale Price	$850
System 2:	Engineering prototype CC-40 with 18K Memory	
	HX-1010 80 Column Thermal Printer + mod to Use Roll Paper	
	HX-3000 RS-232 Interface	
	Sale Price	$400

PC-100C Printer (new) with 6 Rolls of Thermal Paper $150

Write to Maurice Swinnen, 9213 Lanham Severn Road, Lanham MD 20706, or call during working hours (Eastern time) at 301-427-5125. This distribution of material in the two systems is tentative. He is open to suggestions. He will ship by UPS or other carrier per your instructions. You pay the shipping in addition to the prices quoted above.

THE TI-74 - Palmer Hanson. In mid-July the TI-74 became available from Elek-Tek. Page 18 of their Volume 12 catalog shows the following items and prices:

417627	TI-74 BASICALC	$94.00
417635	8K Constant Memory Cartridge	34.00
417643	Printer	69.00
417651	Cassette Interface	22.00
417669	Learn Pascal Software Cartridge	29.00
417677	Statistics Software Cartridge	29.00
417685	Mathematics Software Cartridge	29.00

Those prices are substantially lower than those quoted in V10N4P3 from the Executive Photo Catalog. You can call Elek-Tek toll-free at 800-621-1269 and order by credit card if you like. There is a shipping charge of $4.00 for the first item and an additional $1.00 for each additional item. As of mid-August only the TI-74 and the software cartridges were in stock.

A full size illustration of the TI-74 appears on the front page. It is somewhat smaller than the CC-40, and comes with a snugly fitting plastic carrying case. It will have to be removed from the carrying case to connect to the printer and cassette interface. The advertisement in the Elek-Tek catalog calls the device "An advanced scientific calculator that is BASIC programmable". The CALC mode provides most of the functions which we have come to expect from a scientific calculator from TI including statistics, linear regression, hyperbolic functions, permutations and combinations. The BASIC mode provides many of the functions of the CC-40.

The CALC and BASIC mode both display ten digits and seem to use the same internal mathematics carrying 13 or 14 digits depending upon the number. Thus, in BASIC mode

e = EXP(1) = 2.718281828458 which includes 13 digits, and

e^3 = (EXP(1)^3) = 20.085536923165 which includes 14 digits.

That is generally consistent with Larry Leeds' discussion of the arithmetic of the TI-99/4 and CC-40 in V9N5P6. The "Itsy-bit of Paranoia" test from the February 1985 issue of BYTE (see V10N2P16 for a listing which will run on the TI-74 if each "Print #1" is changed to "Print") yields exactly the same results as the CC-40 with radix = 100, precision = 7, fpwidth = 1.E+14, ulpone = 1.E-14, and the existence of a guard digit for add/subtract.

One of the major deficiencies of the CC-40 keyboard has been removed; the TI-74 has a shift key at the right of the space bar. There are some other keyboard features which are inconvenient. The keyboard is "typewriter-like" for the alphabet and the punctuation symbols, but is entirely too small to be used like a typewriter by any but the smallest hands. To accomodate both the the calculator functions and the Basic functions the upper row of a normal typewriter keyboard (numbers and symbols) has been replaced by a functions row, and numbers and symbols can only be entered from the calculator keypad at the right. One result is that the arrow keys used for editing in Basic, which were directly above the calculator keypad at the right side of the keyboard on the CC-40, have been moved toward the center of the keyboard.

The TI-74 - (cont)

That change is particularly inconvenient for me since the two computers that I used most in the past, the CC-40 and the Model 100, had those keys in very similar positions on the keyboard. However, I have become resigned to the idea that calculator/computer manufacturers change keystrokes and key positions with total disregard for the habits of users. For example, any serious TI-58/59 user had developed a habit of thinking of the Op command as 2nd-Op, the Label command and 2nd-Lbl, etc. as a "crutch" for remembering the keyin sequence for rapid entry of program commands. The TI-66 changed all that, for no apparent good reason so far as I can see.

I tested the TI-74 arithmetic with some of the same benchmarks which we have discussed in earlier issues and obtained mixed results:

1. e x π was equal to π x e indicating that multiplication was commutative. The non-commutative multiply on the TI-59 was discussed in V9N2P15.

2. sin(45) was not equal to cos(45), and in general sin(X) was not equal to cos(90 - X). Page A-33 of the TI-74 Programming Reference Guide makes the helpful suggestion that:

> "A useful technique is to test whether two values are sufficiently close together rather than absolutely equal ...
> ... Instead of IF X=Y THEN ... use
> IF ABS(X-Y) < 1E-11 THEN ... "

That is a safer technique than the idea of doing EE-INV-EE before a comparison as proposed in Personal Programming for the TI-59. Of course the size of the test value depends upon the problem.

3. The square root-squared test: V8N3P13/14 described this test which is a derivative of the $(\sqrt{2})^2$ test by Brian Hayes on page 136 of the January 1981 issue of BYTE. For our test we start with an integer, take the square root five times, take the square five times and compare the result to the original number. I tested selected integers from 2 through 17. The display returned the starting integer in each case. The actual values before truncation to the display were:

2	1.99999 99999 83	12	12.00000 00001 26
3	3.00000 00000 04	13	12.99999 99998 13
5	4.99999 99999 70	15	15.00000 00002 91
7	7.00000 00000 71	17	17.00000 00000 69

where all of the answers are better than the TI-59. Again, note that some solutions show 13 digits and other solutions show 14 digits.

4. 1.0000001 squared 27 times: V9N2P11 described this test from the "Computer Recreations" column of the April 1984 issue of Scientific American, where there are different methods of calculation:

Exact	674530.4707410 84559 ...
Mode A (Repeated A^2)	674530.3180422 5
Mode B (Repeated A*A)	674530.3180422 5
Mode C (A^134217728)	674530.4707401 0

where in tests of other devices only the Model 100 yielded a better answer than the Mode C solution from the TI-74.

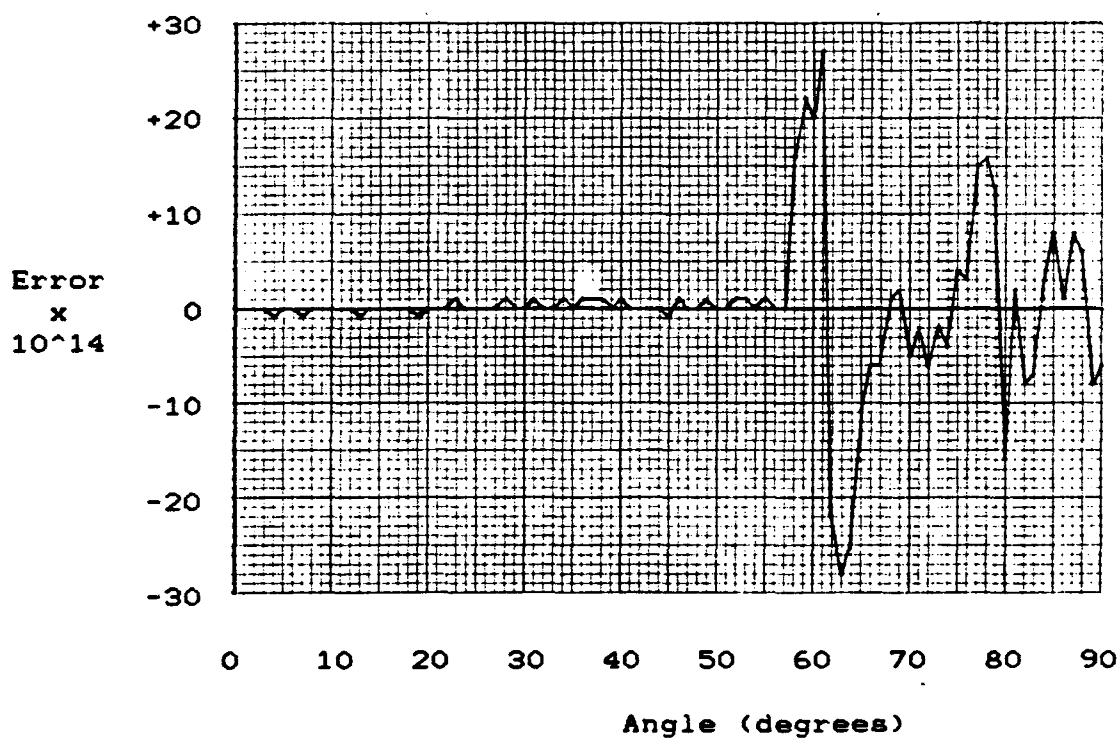
The TI-74 - (cont)

5. The Bob Fruit benchmark: Bob proposed a compound interest problem as another benchmark (see V8N4P4). The appropriate equation is that for the sum of a geometric series S = [(1 + i)^n -1]/i . An annual interest rate of ten per cent (i = .10/12) and compounding monthly for thirty years (n = 360) yields:

Exact	2260.48792 47960 86067 ...
TI-74 using the y^x function	2260.48792 45128

where that answer is better than the CC-40 or TI-99/4, but not as good as the TI-59 or TI-66.

6. Accuracy of the sine function:
The errors in the output of the sine function from the TI-74 are illustrated in the following figure:



For input angles less than one radian (57.2957... degrees) the error is never greater than 1E-14; but for input angles greater than one radian the errors become erratic. For one degree increments over the 0 to 90 degree range the RMS errors was 8E-14 and the peak error was 28E-14. Those values are substantially better than the TI-59, TI-66, or CC-40 and only slightly worse than the Radio Shack Model 100.

It occurred to me that one might be able to extend the very good accuracy for the sine function by using cos(90 - X) for some angles, say for angles above one radian. Examination of the cosine function showed that is not the case. In fact, for the range from 58 degrees through 90 degrees the error in the cos(90 - X) function was exactly the same as the error in the sinX function for 29 of the 33 angles tested. The cosine function is also substantially less accurate than the sine function for angles below one radian with a peak error of 56E-14 and an RMS error of 23E-14 over that range. This suggests that for the most accurate results the user might want to use sin(90 - X) or sqr(1 -sin(X)^2) instead of cos(X) wherever possible.

THE TI-74 RANDOM NUMBER GENERATOR - The RND function provides access to a series of predetermined uniformly distributed pseudo-random numbers with values in the range from 0 to 1. The same sequence of values will be returned each time a program is run. Using a little tallying program to determine the distribution of the RND outputs in ten equal width "buckets" the results for 1000, 10,000 and 100,000 random numbers were

	1000	10,000	100,000
D(1)	113	1,017	9,921
D(2)	95	973	9,999
D(3)	75	958	10,092
D(4)	95	1,013	9,892
D(5)	104	1,008	10,185
D(6)	98	971	9,943
D(7)	105	974	10,044
D(8)	95	1,047	10,031
D(9)	112	1,028	9,994
D(10)	108	1,011	9,899

```
10 FOR I=1 TO 1000
20 A=10*RND+.5
30 D(A)=D(A)+1
40 NEXT I
50 FOR I=1 TO 10
60 PRINT D(I)
70 PAUSE:NEXT I
99 END
```

where the distribution of 1000 numbers is identical to that presented for the CC-40 on V10N1P25. Tests show that the number sequence delivered by the RND function in the TI-74 is identical to that obtained from the CC-40.

The tallying program uses the indirect address sorting technique described in V10N1P24 and V10N3P13. Lines 20 and 30 rely on the characteristic of the TI-74 which selects the address for a non-integer subscript by rounding to the nearest integer. That is another TI-74 feature which is common with the CC-40. As written the program counts into D(1) through D(10). An alternative would have been to write line 20 as A = INT(10*RND) which would count into D(0) through (9). In either case no dimensioning statement is needed since BASIC provides automatic dimensioning of any variable up to 10.

WHAT'S MISSING IN THE TI-74? - The preceding discussion has noted the similarity of the features of the TI-74 to those of the CC-40. There are some attractive features of the CC-40 which were dropped:

* The BEEP which allowed the user to sound a tone as an attention getter, say for erroneous inputs, the end of calculations, etc.

* The CHAR command which allowed the user to define his own set of characters on the CC-40.

* An external power supply to minimize battery useage. Page 1-5 of the TI-74 Users Guide cautions:.

  "The Constant Memory (TM) feature retains stored information for a short time after the batteries are removed. As a precaution, however, you may want to save any important programs and data on a storage device (such as a cassette) before replacing batteries."

* An RS-232 capability.

COMBINATORIAL ANALYSIS ON THE TI-74 - The CALC mode of the TI-74 provides built-in functions for factorials, permutations and combinations. The larger numerical range of the TI-74 (10^127 as opposed to 10^99 with the TI-59, another feature which is common with the CC-40), permits the calculation of factorials up to 84! = 3.314240135E+126. The solution is very fast; the answer seems to appear in the display as soon as the factorial key is pressed. V11N1P5 also reported execution times for benchmark permutation and combination problems. The TI-74 returns the permutations of 100 items taken 50 at a time in about a second, and the combinations of 328 items taken 164 at a time in about 3 seconds. That makes it the fastest hand-held/lap-top device tested so far.

THE STATISTICS LIBRARY CARTRIDGE FOR THE TI-74 - The program package is very similar to that which was available with the Statistics module for the CC-40. Even the examples in the documentation are the same. The major differences are in the responses to prompts. For example, the first prompt with the Means and Moments program asks "Use Printer?", where the acceptable answers are "Y" or "y" for yes and "N" or "n" for no. With the CC-40 the user entered one of those letters and pressed ENTER to proceed to the next step. With the TI-74 the machine moves immediately to the next step on entry of an apropriate response. That should provide some increase in speed of operation; however, at present I am so conditioned to the CC-40 sequence that I press ENTER anyway. The CC-40 sounds a tone when the response to a prompt is not acceptable. The TI-74 can't do that since it does not have a BEEP command.

Page 2-18 of the manual defines the geometric mean as

$$Xg = (X_1^{f_1} \times X_2^{f_2} \times \ldots \times X_K^{f_K})^{1/N} \quad \text{where } N = \sum f_K.$$

That is often written as $Xg = [\prod_{i=1}^{n} X_i^{f_i}]^{1/N}$

The actual Means and Moments routine is probably not mechanized in that way. If it were, you would cause an overflow condition as soon as the product equals or exceeds 10E127. Test cases will show that such overflow does not occur. It seems likely that the geometric mean routine is mechanized using a sum of log(X) formulation as indicated on page 23 of the manual for the Statistics cartridge for the CC-40.

$$Xg = 10\text{^}[(\sum_{i=1}^{n} f_i * \log(x_i))/N]$$

V10N2P22 noted that there was an error in the formula for finding the geometric mean on page 23 of the manual for the Statistics cartridge of the CC-40 and gave a correct formula as shown above. V10N2P3 noted that overflow problems with the calculation of the geometric mean were delayed in the CC-40 because of the log formulation.

THE TI-95 - Page 76 of the June 16, 1986 issue of Electronics includes an announcement of the TI-95 PROCALC (TM). A call to the toll-free 1-800-TI-CARES number revealed that the unit will be available late this fall at a suggested price of $200. The TI-95 did not make the Volume 12 Elek-Tek catalog. You can obtain a brochure by writng to Texas Instruments, Inc., Consumer Marketing, P.O. Box 53, Lubbock, TX 79408.

The brochure shows an actual size photograph which seems to have the same envelope as the TI-74, but with a different layout for keyboard and display. The TI-95 and TI-74 will use the same printer, the PC-324, and the same cassette interface, the CI-7. The Elek-Tek catalog describes the PC-324 as a 24 column, battery-operated thermal printer.

Numeric accuracy will be 13 digits with a 10 digit display. Thirteen digits will be displayed with a special key. Over 200 functions and 251 program codes will be available. The user available internal RAM will be able to be partitioned for up to 900 data registers, 7200 program steps, or 6200 bytes of file space. Optional 8K cartridges will provide more file space, and programs will be able to be run from either the internal memory or the cartridge. Software cartridges will include mathematics, statistics and others.

THE HP-18C - Page 76 of the June 16, 1986 issue of Electronics also included an announcement of the Hewlett-Packard HP-18C Business Consultant calculator. The announcement notes that "... Users of the company's earlier calculators had to program the devices in the RPN language, which stands for Reverse Polish Notation. The Business Consultant features a simplified user interface, algebraic data entry, and built-in programs ...". It has a four line by 23 character LCD display that can displays messages and user prompts. It folds to pocket size. Application booklets and a training guide are available. A printer which will use an infrared interface will be available this fall. The suggested retail price is $175.00, but page 9 of the Volume 12 Elek-Tek catalog offers the HP-18C at $135.00 plus $4.00 for shipping and handling, and the Issue 31A Educalc catalog offers it for $139.95 plus $1.00 for shipping and handling.

The Elek-Tek catalog notes that the device "Uses Industry Standard Algebraic Operating System (Not RPN)". The Educalc catalog notes that "... It calculates with Algebraic Logic, dealing with equations just as you write them." So far I have not seen one in a local showcase, but a dealer in Tampa was kind enough to perform a simple test of the operating system. The sequence 4 + 5 x 6 = yields 54, not the 34 you would expect from the TI A.O.S. The result is consistent with what Lem Matteson called "Adding Machine Logic" where each previous operation is completed because the mechanical adding machines had to operate that way. Adding machine logic is assumed in many business formulas and books. V9N1P4 and V9N3P17 had noted that the TI's BA-55 used adding machine logic. The use of adding machine logic in the BA-55 and HP-18C seems to illustrate the power of the consumer and of established useage.

You can obtain a brochure on the HP-18 from Hewlett-Packard Co., Inquiries Manager, 1000 N.E. Circle Blvd, Corvallis, OR 97330. You can obtain the address of a local distributor by calling toll free 800-FOR-HPPC.

LA GRANGE'S POLYNOMIAL INTERPOLATION - Jorge Valencia of Lima, Peru writes: "At this very late date I have to report an apparent mistake in La Grange's Polynomial Interpolatikon formula as stated in program MA1-06 of the Math Library of the SR-52, issued ten years ago. I was working on development of the formula for a lecture, and going through the algorithm as worked out in the program listing, I realized that it does not follow the formula as stated in the Math Library book:

$$P(x) = \sum_{i=0}^{N} \left[\left(\prod_{\substack{j=0 \\ i \neq j}}^{N} \frac{x - x_i}{x_i - x_j} \right) f(x_i) \right]$$

but rather the formula:

$$P(x) = \sum_{i=0}^{N} \left[\left(\prod_{\substack{j=0 \\ i \neq j}}^{N} \frac{x - x_j}{x_i - x_j} \right) f(x_i) \right]$$

that is to say, that the second term in the numerator of the product is x_j and not x_i. This algorithm gives correct results, whereas the one with x_i does not. Furthermore, the contents of the registers as stated on page 26 of the book are also wrong. R6 contains x , R7 contains x_0, R9 contains y_0 , R9 contains x_1, R10 contains y_1 , and so on."

Editor's Note: It occurred to me that the formulas for the two La Grange Polynomial Interpolation programs for the TI-59, PPX 338004 and 338011, might also be incorrect. Review showed the documentation to be correct for those porgrams.

I looked further at PPX338011 by David Rodabaugh of Columbia, Missouri. The program listing is:

000	76	LBL	022	07	07	044	04	04	066	05	05	088	95	=	110	73	RC*
001	15	E	023	66	PAU	045	08	8	067	65	×	089	55	÷	111	08	08
002	47	CMS	024	43	RCL	046	85	+	068	02	2	090	53	(	112	95	=
003	00	0	025	01	01	047	43	RCL	069	95	=	091	73	RC*	113	44	SUM
004	92	RTN	026	92	RTN	048	03	03	070	42	STO	092	09	09	114	04	04
005	76	LBL	027	76	LBL	049	65	×	071	07	07	093	75	-	115	02	2
006	11	A	028	12	B	050	02	2	072	01	1	094	73	RC*	116	22	INV
007	32	X:T	029	32	X:T	051	95	=	073	42	STO	095	07	07	117	44	SUM
008	01	1	030	09	9	052	42	STO	074	06	06	096	95	=	118	08	08
009	44	SUM	031	61	GTO	053	09	09	075	43	RCL	097	49	PRD	119	22	INV
010	01	01	032	00	00	054	85	+	076	03	03	098	06	06	120	44	SUM
011	08	8	033	12	12	055	01	1	077	32	X:T	099	02	2	121	09	09
012	85	+	034	76	LBL	056	95	=	078	43	RCL	100	22	INV	122	97	DSZ
013	02	2	035	13	C	057	42	STO	079	05	05	101	44	SUM	123	03	03
014	65	×	036	42	STO	058	08	08	080	67	EQ	102	07	07	124	00	00
015	43	RCL	037	02	02	059	43	RCL	081	00	00	103	97	DSZ	125	59	59
016	01	01	038	43	RCL	060	01	01	082	99	99	104	05	05	126	43	RCL
017	95	=	039	01	01	061	42	STO	083	43	RCL	105	00	00	127	04	04
018	42	STO	040	42	STO	062	05	05	084	02	02	106	75	75	128	92	RTN
019	07	07	041	03	03	063	08	8	085	75	-	107	43	RCL	129	00	0
020	32	X:T	042	00	0	064	85	+	086	73	RC*	108	06	06	130	00	0
021	72	ST*	043	42	STO	065	43	RCL	087	07	07	109	65	×	131	00	0

User Instructions:

1. Press E to initialize.

2. Enter the first x value and press A. The value will be flashed, and the calculator stops with a "1." in the display.

3. Enter the first f(x) value and press B. The value will be flashed, and the calculator stops with a "1." in the display. Repeat steps 2 and 3 for as many data pairs as you have. The display will increment to indicate the number of pairs entered.

La Grange's Polynomial Interpolation - (cont)

4. To obtain an interpolated value, enter the x value, and press C. The calculator will run for a while and stop with the f(x) value in the display.

5. You may obtain additional solutions as needed by repeating step 4.

For the fourth degree polynomial interpolation problem defined on page 52 of the manual for the Math/Utilities module, the run time for the solution is 42 seconds. The Aitken interpolation in MU-14 requires about 16 seconds for the same problem.

Rodabaugh's program is exactly the kind of program that can benefit from fast mode. The most straightforward method to obtain fast mode for the solution is to store the x value in R02 and jump to step 038 at entry to fast mode. An appropriate program listing which will add a fast mode option as Lbl C' is

000	76	LBL	027	76	LBL	054	85	+	081	00	00	108	06	06	135	01	1
001	15	E	028	12	B	055	01	1	082	99	99	109	65	×	136	00	0
002	47	CMS	029	32	X:T	056	95	=	083	43	RCL	110	73	RC*	137	69	OP
003	00	0	030	09	9	057	42	STO	084	02	02	111	08	08	138	17	17
004	92	RTN	031	61	GTO	058	08	08	085	75	-	112	95	=	139	71	SBR
005	76	LBL	032	00	00	059	43	RCL	086	73	RC*	113	44	SUM	140	01	01
006	11	A	033	12	12	060	01	01	087	07	07	114	04	04	141	46	46
007	32	X:T	034	76	LBL	061	42	STO	088	95	=	115	02	2	142	32	X:T
008	01	1	035	13	C	062	05	05	089	55	÷	116	22	INV	143	25	CLR
009	44	SUM	036	42	STO	063	08	8	090	53	(	117	44	SUM	144	32	X:T
010	01	01	037	02	02	064	85	+	091	73	RC*	118	08	08	145	91	R/S
011	08	8	038	43	RCL	065	43	RCL	092	09	09	119	22	INV	146	22	INV
012	85	+	039	01	01	066	05	05	093	75	-	120	44	SUM	147	57	ENG
013	02	2	040	42	STO	067	65	×	094	73	RC*	121	09	09	148	04	4
014	65	×	041	03	03	068	02	2	095	07	07	122	97	DSZ	149	05	5
015	43	RCL	042	00	0	069	95	=	096	95	=	123	03	03	150	02	2
016	01	01	043	42	STO	070	42	STO	097	49	PRD	124	00	00	151	85	+
017	95	=	044	04	04	071	07	07	098	06	06	125	59	59	152	01	1
018	42	STO	045	08	8	072	01	1	099	02	2	126	43	RCL	153	52	EE
019	07	07	046	85	+	073	42	STO	100	22	INV	127	04	04	154	01	1
020	32	X:T	047	43	RCL	074	06	06	101	44	SUM	128	92	RTN	155	02	2
021	72	ST*	048	03	03	075	43	RCL	102	07	07	129	00	0	156	95	=
022	07	07	049	65	×	076	03	03	103	97	DSZ	130	00	0	157	22	INV
023	66	PAU	050	02	2	077	32	X:T	104	05	05	131	76	LBL	158	52	EE
024	43	RCL	051	95	=	078	43	RCL	105	00	00	132	18	C'	159	86	STF
025	01	01	052	42	STO	079	05	05	106	75	75	133	42	STO			
026	92	RTN	053	09	09	080	67	EQ	107	43	RCL	134	02	02			

where the fast mode entry technique was discussed in detail in V9N1P15 and V9N2P3. Steps 135 through 139 ensure that the partitioning is set properly for the end-of-partition Stflg. Steps 139 through 141 store a subroutine return address which will control the exit from fast mode by the RTN at step 128. Steps 142 through 145 clear the error condition set at fast mode entry before the program stops with the solution in the display. The 45 at steps 148 through 149 which sets the jump address at fast mode entry to step 038 may be found from the Carl Rabe's table on V9N3P5, or may be calculated from the formula

$$\text{Jump address} = 8*(WXY) + Z + 1$$

where W, X, Y, and Z are the ninth through twelfth digits of the fast mode entry constant. For our problem, X = 0, Y = 4 and Z = 5 to make the jump address 8*4 + 5 + 1 = 38. The user instructions are the same as for the PPX program if you wish to run in normal mode. For fast mode operation, step 4 is changed to "... enter the x value and press C'. See a flashing "1. 12" in the display and press 7 and then EE. ...". For the problem from the M/U manual the solution will be obtained in about 23 seconds.

La Grange's Polynomial Interpolation - (cont)

The fast mode program on page 11 accomplishes the purpose of speeding up the solution. However, additional changes will provide a printout of the input values together with the number of the input pair, and a printout of the solution. The program is rearranged to permit use of the jump to step 001 technique of fast mode entry. The program listing appears below. The instructions are the same as for the PPX program except that step 4 is changed to " ... enter the x value and press C. The calculator will stop with a flashing "1." in the display. Press 7 and then EE to begin the solution in fast mode. ... ". A copy of the printout which will appear for the problem from page 52 of the M/U manual appears at the right.

1.	N
0.6	X
0.2257	Y
2.	N
0.7	X
0.258	Y
3.	N
0.9	X
0.3159	Y
4.	N
1.	X
0.3413	Y
5.	N
1.1	X
0.3643	Y
0.8	X
0.28811	Y

000	91	R/S	040	03	03	080	22	INV	120	69	DP	160	00	0	200	00	0
001	25	CLR	041	32	X:T	081	44	SUM	121	04	04	161	00	0	201	00	0
002	43	RCL	042	43	RCL	082	08	08	122	08	8	162	00	0	202	00	0
003	01	01	043	05	05	083	22	INV	123	85	+	163	00	0	203	00	0
004	42	STD	044	67	EQ	084	44	SUM	124	02	2	164	00	0	204	00	0
005	03	03	045	00	00	085	09	09	125	65	×	165	00	0	205	00	0
006	00	0	046	63	63	086	97	DSZ	126	43	RCL	166	00	0	206	00	0
007	42	STD	047	43	RCL	087	03	03	127	01	01	167	00	0	207	00	0
008	04	04	048	02	02	088	00	00	128	95	=	168	00	0	208	00	0
009	08	8	049	75	-	089	23	23	129	42	STD	169	00	0	209	00	0
010	85	+	050	73	RC*	090	04	4	130	07	07	170	00	0	210	00	0
011	43	RCL	051	07	07	091	05	5	131	32	X:T	171	00	0	211	00	0
012	03	03	052	95	=	092	69	DP	132	69	DP	172	00	0	212	00	0
013	65	×	053	55	÷	093	04	04	133	06	06	173	00	0	213	00	0
014	02	2	054	53	(	094	43	RCL	134	72	ST*	174	00	0	214	00	0
015	95	=	055	73	RC*	095	04	04	135	07	07	175	00	0	215	00	0
016	42	STD	056	09	09	096	69	DP	136	92	RTN	176	00	0	216	76	LBL
017	09	09	057	75	-	097	06	06	137	76	LBL	177	00	0	217	13	C
018	85	+	058	73	RC*	098	81	RST	138	12	B	178	00	0	218	42	STD
019	01	1	059	07	07	099	76	LBL	139	32	X:T	179	00	0	219	02	02
020	95	=	060	95	=	100	15	E	140	04	4	180	00	0	220	32	X:T
021	42	STD	061	49	PRD	101	47	CMS	141	05	5	181	00	0	221	98	ADV
022	08	08	062	06	06	102	25	CLR	142	69	DP	182	00	0	222	09	9
023	43	RCL	063	02	2	103	91	R/S	143	04	04	183	00	0	223	69	DP
024	01	01	064	22	INV	104	76	LBL	144	09	9	184	00	0	224	17	17
025	42	STD	065	44	SUM	105	11	A	145	61	GTD	185	00	0	225	60	DEG
026	05	05	066	07	07	106	32	X:T	146	01	01	186	00	0	226	22	INV
027	08	8	067	97	DSZ	107	69	DP	147	23	23	187	00	0	227	57	ENG
028	85	+	068	05	05	108	21	21	148	00	0	188	00	0	228	04	4
029	43	RCL	069	00	00	109	03	3	149	00	0	189	00	0	229	04	4
030	05	05	070	39	39	110	01	1	150	00	0	190	00	0	230	69	DP
031	65	×	071	43	RCL	111	69	DP	151	00	0	191	00	0	231	04	04
032	02	2	072	06	06	112	04	04	152	00	0	192	00	0	232	32	X:T
033	95	=	073	65	×	113	98	ADV	153	00	0	193	00	0	233	69	DP
034	42	STD	074	73	RC*	114	43	RCL	154	00	0	194	00	0	234	06	06
035	07	07	075	08	08	115	01	01	155	00	0	195	00	0	235	04	4
036	01	1	076	95	=	116	69	DP	156	00	0	196	00	0	236	05	5
037	42	STD	077	44	SUM	117	06	06	157	00	0	197	00	0	237	30	TAN
038	06	06	078	04	04	118	04	4	158	00	0	198	00	0	238	33	X^2
039	43	RCL	079	02	2	119	04	4	159	00	0	199	00	0	239	86	STF

TI-55II REPLACEMENTS - V9N6P2 and V10N1P2 discussed problems with key bounce on the TI-55II and other LCD calculators. V11N1P13 noted that the TI-55III keyboard has a different feel, and that hopefuly the key bounce problems have been solved. One member wrote to tell me that there is a TI-55II replacement policy. I called 1-800-TI-CARES. I gave them the serial number and date code of my TI-55II together with my name and address. They will send a replacement.

In only nine days from the telephone call a TI-55III arrived. It cost $1.48 to return the defective TI-55II. As the telephone number says, "TI CARES".

KEY CODE 46 - V5N8P2 reported that a synthesized key code 46 (INS) could be used instead of a CE as a dummy operator to move a quantity inside an opening parenthesis. That useage has the advantage that it does not clear an existing error condition. The article "Keeping It Simple" in the November/December issue of PPX Exchange contains the following discussion:

> "KEY CODE 46? Several members have noted the existence of certain key codes in a program which have no corresponding key sequence in the key code table as listed on page V-50 of the Personal Programming manual. For instance, the key codes for [GTO] 146 would appear in the program listing as [61] [01] [46]. When encountered during program execution, control would be transferred to program location 146. ... "

One place where the PPX members might have encountered code 46 was in my fast mode calendar printer program (PPX 908192G). I had commented on the unusual useage in the program description, and had also noted in the program listing that code 46 "Does the CE function without resetting error conditions". Obviously, the PPX editor was not familiar with the use of code 46 (INS) as a dummy operator. A similar question from Peter Stromgren of Denmark was answered in V9N3P10.

I trust that readers of TI PPC Notes are not confused by this useage which appears in some programs.

MAILBAG

"I should like confirmation that the last issue of PPX Exchange was Volume 6 Number 6 for November-December 1982. ... ... Keep up the good work. What would we do without you?". J.V.

> V6N6 was the last issue. In mid to late January 1983 you should have received a letter from TI notifying you of the decision to discontinue the PPX-59 club by the end of the first quarter of 1983, and extending a $20.00 credit toward the selection of five PPX programs. At the time I thought the letter meant that we would get at least one issue of PPX Exchange in early 1983, but that turned out not to be the case.

"I still read the "Notes" with interest even though I am not using my TI-59 as much as I used to. I have an IBM XT." S.H.

"Would like to see news and reviews of the new TI-74 BASICALC. Looks like a great bargain to me." M.F.

"... I particularly like the program exchange and coverage of other programmable calculators." L.P.

"Keep up the TI coverage, especially the 58C, 59, 66 and 74." W.M.

"I have an FX-7000G and like it, but it is not a 59. I miss the register operation keys and the indirect capabilities. However, the calculator nut should not fail to try one." G.F.

"Has anyone come up with a TI-59 program to simulate a Smith Chart?"

> The abstract for PPX program 648023B states that is simulates the use of a Smith Chart.

LOAN SCHEDULE - Walter Bodenmuller of Calgary, Alberta, Canada called my attention to this old program by Lem Matteson which calculates loan schedules for multiple years. The program as it appeared in V6N9/10P24/25 is listed on page 15 together with a sample printout. The Pgm-02-SBR-239 method of fast mode entry is very dependent on a good card reader and also erases user memory. Lem's program accepts the parameters for the loan in normal mode, stores them in the hierarchy registers (steps 448-482), and recalls them after fast mode entry (steps 027-038).

The conversion of the program to use the Stflg at the end of the partitioning provides another example of how easily that can be accomplished if the fast mode portion is already written using direct addresses and without subroutines. I also found that an INV Stflg 01 was needed if a second loan was to be printed out properly. The changed code is listed below.

```
000  05  5        234  98 ADV      483  61 GTO      548  02  2
001  69 OP        235  98 ADV      484  00  00      549  01  1
002  17  17       236  98 ADV      485  00  00      550  02  2
003  22 INV       237  25 CLR      486  00  0       551  85  +
004  86 STF       238  92 RTN      487  00  0       552  01  1
005  01  01       239  00  0                        553  52 EE
006  87 IFF                                         554  01  1
007  02  02                                         555  02  2
008  00  00                                         556  95  =
009  19  19                                         557  22 INV
010  71 SBR                                         558  52 EE
011  05  05                                         559  86 STF
012  48  48
013  25 CLR
014  91 R/S
015  00  0
016  00  0
017  00  0
018  25 CLR
```

The user instructions are:

1. Enter the amount of the loan and press A.

2. Enter the annual interest rate (%) and press B.

3. Optional - If you want the starting month to be other than January then enter the starting month (1 through 12) and press 2nd A'.

4. Enter the number of years of the loan and press C. You can enter fractional years as shown in the example at the right.

5. Optional - In step 4 the program computes and prints a monthly payment. If you want to use another monthly payment, enter the amount and press D. The program will display and print the number of payments needed with this new payment.

6a. To print the schedule in normal mode press E. A one year schedule will take about 67 seconds.

6b. To print the schedule in fast mode press 2nd E'. The "SCHEDULE" heading will be printed and a flashing "1. 12" will appear in the display. Press 7 and then EE. A one year schedule will take about 36 seconds.

```
     1000.00      LOAN
        12.5      %INT
          8.       MTH

        0.75       YRS
          9.       NO.
      116.98       PMT

      SCHEDULE

     1000.00      LOAN
        12.5      %INT
      116.98       PMT

          1.       NO.
       10.42       INT
      106.56      PCPL
      893.44       BAL

          2.       NO.
        9.31       INT
      107.67      PCPL
      785.77       BAL

          3.       NO.
        8.19       INT
      108.79      PCPL
      676.98       BAL

          4.       NO.
        7.05       INT
      109.93      PCPL
      567.05       BAL

          5.       NO.
        5.91       INT
      111.07      PCPL
      455.98       BAL

    YEARS   INTEREST
     40.88

          6.       NO.
        4.75       INT
      112.23      PCPL
      343.75       BAL

          7.       NO.
        3.58       INT
      113.40      PCPL
      230.35       BAL

          8.       NO.
        2.40       INT
      114.58      PCPL
      115.77       BAL

          9.       NO.
        1.21       INT
      115.77      PCPL
        0.00       BAL

    YEARS   INTEREST
     11.94

    TOTAL   INTEREST
     52.82
```

LOAN SCHEDULE - (cont) - Old Program Listing from V6N9/10P25

```
56000.00   LOAN
17.5       %INT
1.         MTH

30.        YRS
360.       NO.
821.14     PMT

821.14     PMT
360.03     NO.

SCHEDULE

56000.00   LOAN
17.5       %INT
821.14     PMT

1.         NO.
816.67     INT
4.47       PCPL
55995.53   BAL

2.         NO.
816.60     INT
4.54       PCPL
55990.99   BAL

3.         NO.
816.54     INT
4.60       PCPL
55986.39   BAL

4.         NO.
816.47     INT
4.67       PCPL
55981.72   BAL

5.         NO.
816.40     INT
4.74       PCPL
55976.98   BAL

6.         NO.
816.33     INT
4.81       PCPL
55972.17   BAL

7.         NO.
816.26     INT
4.88       PCPL
55967.29   BAL

8.         NO.
816.19     INT
4.95       PCPL
55962.34   BAL

9.         NO.
816.12     INT
5.02       PCPL
55957.32   BAL

10.        NO.
816.04     INT
5.10       PCPL
55952.22   BAL

11.        NO.
815.97     INT
5.17       PCPL
55947.05   BAL

12.        NO.
815.89     INT
5.25       PCPL
55941.80   BAL

YEARS   INTEREST
9795.48

13.        NO.
815.82     INT
5.32       PCPL
55936.48   BAL
```

```
453536.       30
303723.       31
61243137.     32
27321331.     33
4517133536.   34
3732371327.   35
243137.       36
1735173637.   37
361523.       38
1716412717.   39
0.            40
0.            41
0.            42
313240.       43
333037.       44
33153327.     45
141327.       46
```

```
000  00  0
001  00  0
002  00  0
003  00  0
004  00  0
005  36  PGM
006  02  02
007  71  SBR
008  02  02
009  39  39
010  00  0
011  00  0
012  00  0
013  00  0
014  00  0
015  00  0
016  00  0
017  25  CLR
018  91  R/S
-------------
019  22  INV
020  58  FIX
021  22  INV
022  86  STF
023  02  02
024  05  5
025  69  OP
026  17  17
027  82  HIR
028  16  16
029  42  STO
030  28  28
031  82  HIR
032  17  17
033  42  STO
034  26  26
035  82  HIR
036  12  12
037  42  STO
038  02  02
039  00  0
040  42  STO
041  01  01
042  42  STO
043  23  23
044  42  STO
045  24  24
046  43  RCL
047  33  33
048  69  OP
049  04  04
050  58  FIX
051  02  02
052  82  HIR
053  14  14
054  69  OP
055  06  06
056  22  INV
057  58  FIX
058  43  RCL
059  32  32
060  69  OP
061  04  04
062  43  RCL
063  26  26
064  69  OP
065  06  06
066  43  RCL
067  44  44
068  69  OP
069  04  04
070  58  FIX
071  02  02
072  82  HIR
073  13  13
074  69  OP
075  06  06
076  22  INV
077  58  FIX
078  98  ADV
079  82  HIR
080  15  15
081  59  INT
082  42  STO
083  00  00
084  43  RCL
085  43  43
086  69  OP
087  04  04
088  69  OP
089  21  21
090  43  RCL
091  01  01
092  69  OP
093  06  06
094  43  RCL
095  36  36
096  69  OP
097  04  04
098  82  HIR
099  14  14
100  65  x
101  43  RCL
102  28  28
103  95  =
104  58  FIX
105  02  02
106  52  EE
107  22  INV
108  52  EE
109  69  OP
110  06  06
111  22  INV
112  58  FIX
113  44  SUM
114  23  23
115  44  SUM
116  24  24
117  32  X:T
118  43  RCL
119  45  45
120  69  OP
121  04  04
122  87  IFF
123  01  01
124  01  01
125  80  80
126  82  HIR
127  13  13
128  75  -
129  32  X:T
130  95  =
131  58  FIX
132  02  02
133  69  OP
134  06  06
135  22  INV
136  58  FIX
137  82  HIR
138  54  54
139  43  RCL
140  46  46
141  69  OP
142  04  04
143  58  FIX
144  02  02
145  82  HIR
146  14  14
147  69  OP
148  06  06
149  22  INV
150  58  FIX
151  98  ADV
152  87  IFF
153  01  01
154  04  04
155  88  88
156  22  INV
157  97  DSZ
158  02  02
159  04  04
160  88  88
161  97  DSZ
162  00  00
163  00  00
164  84  84
165  86  STF
166  01  01
167  82  HIR
168  14  14
169  32  X:T
170  00  0
171  67  EQ
172  04  04
173  88  88
174  77  GE
175  04  04
176  88  88
177  61  GTO
178  00  00
179  84  84
-------------
180  82  HIR
181  14  14
182  85  +
183  69  OP
184  06  06
185  32  X:T
186  95  =
187  82  HIR
188  03  03
189  43  RCL
190  44  44
191  69  OP
192  04  04
193  58  FIX
194  02  02
195  82  HIR
196  13  13
197  69  OP
198  06  06
199  22  INV
200  58  FIX
201  00  0
202  82  HIR
203  04  04
204  61  GTO
205  01  01
206  39  39
-------------
207  43  RCL
208  35  35
209  69  OP
210  02  02
211  43  RCL
212  36  36
213  69  OP
214  03  03
215  43  RCL
216  37  37
217  69  OP
218  04  04
219  69  OP
220  05  05
221  58  FIX
222  02  02
223  43  RCL
224  24  24
225  99  PRT
226  22  INV
227  58  FIX
228  22  INV
229  86  STF
230  00  00
231  06  6
232  69  OP
233  17  17
234  98  ADV
235  98  ADV
236  98  ADV
237  25  CLR
238  91  R/S
-------------
239  00  0
240  76  LBL
241  16  A'
242  22  INV
243  58  FIX
244  32  X:T
245  05  5
246  69  OP
247  17  17
248  61  GTO
249  05  05
250  24  24
-------------
251  76  LBL
252  15  E
253  86  STF
254  02  02
255  61  GTO
256  10  E'
-------------
257  76  LBL
258  11  A
259  42  STO
260  29  29
261  05  5
262  69  OP
263  17  17
264  43  RCL
265  33  33
266  69  OP
267  04  04
268  87  IFF
269  03  03
270  02  02
271  76  76
272  01  1
273  02  2
274  42  STO
275  02  02
276  58  FIX
277  02  02
278  43  RCL
279  29  29
280  69  OP
281  06  06
282  91  R/S
-------------
283  76  LBL
284  12  B
285  22  INV
286  58  FIX
287  42  STO
288  26  26
289  55  ÷
290  01  1
291  02  2
292  00  0
293  00  0
294  95  =
295  42  STO
296  28  28
297  43  RCL
298  32  32
299  69  OP
300  04  04
301  43  RCL
302  26  26
303  69  OP
304  06  06
305  91  R/S
-------------
306  76  LBL
307  13  C
308  98  ADV
309  32  X:T
310  43  RCL
311  30  30
312  69  OP
313  04  04
314  32  X:T
315  69  OP
316  06  06
317  65  x
318  01  1
319  02  2
320  95  =
321  58  FIX
322  00  00
323  88  DMS
324  22  INV
325  58  FIX
326  42  STO
327  20  20
328  43  RCL
329  43  43
330  69  OP
331  04  04
332  43  RCL
333  20  20
334  69  OP
335  06  06
336  59  INT
337  82  HIR
338  05  05
339  43  RCL
340  29  29
341  65  x
342  53  (
343  43  RCL
344  28  28
345  65  x
346  53  (
347  01  1
348  75  -
349  53  (
350  01  1
351  85  +
352  43  RCL
353  28  28
354  54  )
355  45  Yx
356  82  HIR
357  15  15
358  94  +/-
359  54  )
360  35  1/X
361  95  =
362  58  FIX
363  02  02
364  52  EE
365  22  INV
366  52  EE
367  42  STO
368  27  27
369  43  RCL
370  44  44
371  69  OP
372  04  04
373  43  RCL
374  27  27
375  69  OP
376  06  06
377  91  R/S
-------------
378  76  LBL
379  14  D
380  98  ADV
381  42  STO
382  27  27
383  43  RCL
384  44  44
385  69  OP
386  04  04
387  58  FIX
388  02  02
389  43  RCL
390  27  27
391  69  OP
392  06  06
393  22  INV
394  58  FIX
395  01  1
396  85  +
397  43  RCL
398  28  28
399  95  =
400  23  LNX
401  35  1/X
402  65  x
403  53  (
404  01  1
405  75  -
406  43  RCL
407  29  29
408  65  x
409  53  (
410  43  RCL
411  28  28
412  55  ÷
413  43  RCL
414  27  27
415  54  )
416  54  )
417  23  LNX
418  94  +/-
419  95  =
420  58  FIX
421  02  02
422  52  EE
423  22  INV
424  52  EE
425  42  STO
426  20  20
427  59  INT
428  82  HIR
429  05  05
430  22  INV
431  58  FIX
432  43  RCL
433  43  43
434  69  OP
435  04  04
436  43  RCL
437  20  20
438  69  OP
439  06  06
440  91  R/S
-------------
441  76  LBL
442  10  E'
443  22  INV
444  58  FIX
445  22  INV
446  86  STF
447  03  03
448  43  RCL
449  02  02
450  82  HIR
451  02  02
452  98  ADV
453  00  0
454  69  OP
455  04  04
456  43  RCL
457  38  38
458  69  OP
459  02  02
460  43  RCL
461  39  39
462  69  OP
463  03  03
464  69  OP
465  05  05
466  98  ADV
467  43  RCL
468  29  29
469  82  HIR
470  04  04
471  43  RCL
472  27  27
473  82  HIR
474  03  03
475  43  RCL
476  28  28
477  82  HIR
478  06  06
479  43  RCL
480  26  26
481  82  HIR
482  07  07
483  87  IFF
484  02  02
485  00  00
486  19  19
487  81  RST
-------------
488  43  RCL
489  34  34
490  69  OP
491  02  02
492  43  RCL
493  36  36
494  69  OP
495  03  03
496  43  RCL
497  37  37
498  69  OP
499  04  04
500  69  OP
501  05  05
502  58  FIX
503  02  02
504  43  RCL
505  23  23
506  99  PRT
507  22  INV
508  58  FIX
509  98  ADV
510  87  IFF
511  01  01
512  02  02
513  07  07
514  00  0
515  42  STO
516  23  23
517  01  1
518  02  2
519  42  STO
520  02  02
521  61  GTO
522  01  01
523  61  61
524  86  STF
525  03  03
526  43  RCL
527  31  31
528  69  OP
529  04  04
530  32  X:T
531  42  STO
532  03  03
533  69  OP
534  06  06
535  94  +/-
536  85  +
537  01  1
538  03  3
539  95  =
540  42  STO
541  02  02
542  43  RCL
543  03  03
544  91  R/S
-------------
545  25  CLR
546  91  R/S
-------------
```

Loan Schedule - (cont)

The modification process to incorporate the Stflg at the end-of-partition method of fast mode entry is efficient--it only took about ten minutes to complete for this program. The resulting prograqm is anything but elegant. There are three 5-Op-17 sequences where one should do, the temporary storage of the loan parameters into the hierarchy registers is no longer needed, and the various INV Stflg commands could probably be replaced by one appropriately placed RST. We might be able "optimize" the program to fit in 480 steps, but to what end? The resulting program would still use three magnetic card sides.

```
   86.47
   87.14

     12.      NO.
    0.73      INT
   87.19     PCPL
   -0.05      BAL

YEARS  INTEREST
   54.99

YEARS  INTEREST
    0.00

TOTAL  INTEREST
   54.99
```

V6N9/10P24 stated "...if you entered a different payment through D, the program computes a partial last payment. Otherwise, it prints a zero payment as the last payment." The printouts of the last months and the summary at the right show that the program will sometimes print a small partial last payment. The upper printout is for a principal of $1000, an interest rate of 10% and one year. The lower printout is fora principal of $2000, an interest rate of 5%, and one year. The upper print-out includes a second "Years Interest" summary for no apparent good reason, and ends with a small negative balance. The lower printout adds a 13th month to take care of the small monthly balance at the end of twelve months. Both effects seem to be the result of roundoff to the nearest penny in the calculations. One could add some routines to modifiy the payment at the 12th month to eliminate these idiosyncrasies, but again, to what end?

```
  169.
  170.54

     12.      NO.
    0.71      INT
  170.50     PCPL
    0.04      BAL

YEARS  INTEREST
   54.56

     13.      NO.
    0.00      INT
    0.04     PCPL
    0.04      PMT
    0.00      BAL

YEARS  INTEREST
    0.00

TOTAL  INTEREST
   54.56
```

MORE MEMBERSHIP LISTING

V10N4P9 and V11N1P7 listed the names and addresses of members who gave permission to publish that information. Four additional members have agreed to publish similar information:

Dr. Haddad Menashe
7 Dangur Street
Ramat-Gan 52260
Israel

William H. Bowen
10111 Bissonet St. #170
Houston TX 77036-7816
(New address)

Jose M. Gallego
250 Quintard St. Apt. 96
Chula Vista CA 92011

Major Wallace E. Mitchell
P.O. Box 423
Hampstead NH 03841

David M. Douglas, P.E.
6400 Center Street #76
Mentor OH 44060

MORE HARDWARE

I have an AC adapter from a TI-58/59 (6 VAC, 175 ma) which I would be glad to send to anyone who wants it. P. G. Manney, 1166 Lafayette Road, #G-28, Medina OH 44256.

IT'S CALL SQ(R,I) NOT CALL SQR(R,I) FOR THE CC-40 & TI-74 MATH MODULES

Page 103 of the manual for Mathematics Cartridge for the CC-40 shows that CALL SQR(REAL,IMAGINARY) allows the user to pass in the real and imaginary parts of a complex number and obtain the real and imaginary parts of the square of the input. Page A-19 of the manual for the Mathematics cartridge for the TI-74 says the same. However, if you write a short demonstration program:

```
10 INPUT A,B      20 CALL SQR(A,B)      30 PRINT A,B:PAUSE
```

you will find that you get an error message "Program not found" on the CC-40 and "E13 at 20 not found" on the TI-74. The solution is to change the call to CALL SQ(A,B), that is to delete the R. Then, an input of 3,4 returns the expected -7,24.

MORE TI-74 AVAILABILITY - Page 56 of issue 32 of the Educalc Catalog offers TI-74 hardware at the following prices:

TI-74	BASICALC	$109.95
PC-324	Portable Printer	89.95
AC-9201	Adapter	8.95
TP-324	Printer Paper	4.95
CI-7	Cassette Interface	26.95
74-695	8K RAM Cartridge	39.95
74-696	Learn PASCAL Cartridge	36.95
74-697	Statistics Cartridge	36.95
74-698	Mathematics Cartridge	36.95

where the AC-9201 has not been listed in other TI-74 literature. The AC -9201 was used with the CC-40 which had an AC adapter socket. My TI-74 does not have an adapter socket. Perhaps the AC-9201 is used with the printer or the cassette interface. I have not yet been able to obtain those units. You can order with Mastercard or Visa from Educalc by calling 714-582-2637 or toll-free 1-800-633-2252, extension 342. There is a one dollar shipping/handling charge per order.

ANOTHER SPECIALTY PAKETTE AVAILABLE FOR LOAN - Member Gene Freil sent in the Lab Chemistry Specialty Pakette to be available for loan. That pakette includes six PPX programs:

418001A	Perfect Gas Law	418015C	General Thermodynamics
418006B	Elemental Composition	418016C	Aq Acid/Base Buffer Eq
418014C	Least Sq Activation Energy	668009B	Psychometric Calculator

If you want to borrow this pakette send three dollars. I will send it first class and expect you to return it the same way.

ONE SET OF USED 1980/1981 ISSUES - New members who have inquired about the availability of back issues are aware that I have noted that only Xerox copies are available for 1980 and 1981, and some marginal legibility will result. A complete set of old issues was donated to the club by William Vogel (see V11N1P2 and V11N1P23). The 1980 and 1981 issues are somewhat dog-eared and include some of William's notes, but are as legible as any of these issues can be. If you would like the set of used 1980/1981 issues send twenty dollars (thirty dollars overseas) and I will send them to you. First come, first served.

STUDENT'S t-DISTRIBUTION VALUES — *Jorge Valencia.* Program 21 of the TI-59 Applied Statistics Library (ST-2) gives confidence levels for a one-tailed arrangement of Student's t-distribution through input of the degrees of freedom (ν) and the values of t. Confidence level is the cumulative function

$$P(t) = \int_{-\infty}^{t} f(u)\,du$$

where

$$f(u) = \frac{\{\Gamma[(\nu+1)/2]\}\ [1+(u^2/\nu)]^{-(\nu+1)/2}}{\sqrt{\nu\pi}\ \Gamma(\nu/2)}$$

The value of t for a one-tailed distribution — and sometimes also for a two-tailed distribution — is found on tables in most statistics textbooks for the usually required confidence levels: 90%, 95%, 99%, 99.5%, plus a few other, lower levels.

Our "Student's t-Distribution Values" program does the opposite as Pgm 21 of ST-2, and further: by entering the degrees of freedom and the confidence level, it finds the t-value for one- and two-tailed arrangements. The method used is Newton-Raphson iteration on values found through routines in ST-2. *This means that it requires Module ST-2.*

Using this program has several advantages:

1. It calculates t-values to a greater degree of accuracy than can be found on tables.
2. It obviates having to find equivalence between one- and two-tailed confidence levels when one type of table is available and the other type is needed.
3. It finds t-values for degrees of freedom and for confidence levels that are not found in tables. Doing this otherwise would mean reckoning gamma functions and then developing the integral, which is a chore to program and carry out, even with a TI-59 or a larger computer.

User instructions are as follows:

1. Insert Applied Statistics module.
2. Enter program.
3. Initialize. Press E for a one-tailed level (a figure 1 is displayed), or 2nd E' for a two-tailed level (a figure 2 is displayed).
4. Enter degrees of freedom: press A. Input is returned in display.
5. Enter confidence level as a percentage; e.g.: 95 for 95%: press B. Input is returned in display.
6. Find t-value: press C. Value is displayed with three decimal places, as on most tables.
7. For more decimal places, press INV 2nd Fix.
8. For more accuracy, change figure in step 050 from 5 to a larger figure, up to 9.

Editor's Note: For a one-tailed level with the degrees of freedom at 20 and the confidence level at 95% the program yields the answer 1.725 in 2 minutes 30 seconds. It might seem that this program is a good candidate for conversion to fast mode; unfortunately, my tests show that about 2 of the 2½ minutes are used in the calls to the module in subroutine D, and it is well-known that the TI-59 is already operating in fast mode when running a program from a solid state module. Does anyone have another, faster solution?

Student's t-distribution Values - (cont)

000	91	R/S	023	45	Y^x	046	76	LBL	069	75	-	092	43	RCL	115	43	RCL
001	76	LBL	024	32	X:T	047	13	C	070	52	EE	093	09	09	116	09	09
002	14	D	025	01	1	048	01	1	071	06	6	094	65	×	117	22	INV
003	53	(	026	54	)	049	52	EE	072	94	+/-	095	02	2	118	52	EE
004	24	CE	027	55	÷	050	05	5	073	54	)	096	52	EE	119	58	FIX
005	36	PGM	028	01	1	051	94	+/-	074	54	)	097	06	6	120	03	03
006	21	21	029	00	0	052	42	STO	075	14	D	098	94	+/-	121	60	DEG
007	15	E	030	00	0	053	08	08	076	75	-	099	54	)	122	81	RST
008	75	-	031	87	IFF	054	02	2	077	53	(	100	44	SUM	123	76	LBL
009	43	RCL	032	00	00	055	42	STO	078	43	RCL	101	09	09	124	15	E
010	01	01	033	00	00	056	09	09	079	09	09	102	29	CP	125	86	STF
011	54	)	034	40	40	057	53	(	080	65	×	103	77	GE	126	00	00
012	92	RTN	035	85	+	058	43	RCL	081	53	(	104	01	01	127	76	LBL
013	76	LBL	036	01	1	059	09	09	082	01	1	105	07	07	128	10	E'
014	11	A	037	95	=	060	14	D	083	85	+	106	94	+/-	129	47	CMS
015	36	PGM	038	55	÷	061	55	÷	084	52	EE	107	75	-	130	87	IFF
016	21	21	039	02	2	062	53	(	085	06	6	108	43	RCL	131	00	00
017	11	A	040	95	=	063	53	(	086	94	+/-	109	08	08	132	01	01
018	43	RCL	041	42	STO	064	43	RCL	087	54	)	110	95	=	133	36	36
019	15	15	042	01	01	065	09	09	088	54	)	111	29	CP	134	02	2
020	92	RTN	043	25	CLR	066	65	×	089	14	D	112	77	GE	135	92	RTN
021	76	LBL	044	32	X:T	067	53	(	090	54	)	113	00	00	136	01	1
022	12	B	045	92	RTN	068	01	1	091	65	×	114	57	57	137	92	RTN

A RANDOM WALK DEMONSTRATION WITH THE fx-7000G - Palmer Hanson. The idea of "random walk" is frequently encountered in my work in inertial navigation. The Feynman Lectures on Physics (Addison-Wesley) note that

"In its simplest version, we imagine a 'game' in which a 'player' starts at the point x = 0 and at each 'move' is required to take a step either forward (toward +x) or backward (toward -x). The choice is to be made randomly, determined, for example, by the toss of a coin ..."

Analysis shows that the expected behavior is such that the square of the distance traveled in N steps is N, where the expected value is the most probable value. More complex random walk phenomena involves both the direction and the magnitude of each step determined randomly.

The display of the fx-7000G provides an excellent vehicle for a demonstration of random walk. The program at the right mechanizes the simpler random walk idea, where the lengths of the steps are all the same. Line 5 decides whether to step forward or back based on whether the random number generator output is less than 0.58, not the 0.5 that might be expected. That is a result of the non-uniform distribution of the output of the Ran# function of the fx-7000G as described in V11N1P9. Line 8 deletes the x value which would otherwise appear at the lower left portion of the screen.

```
1  Lbl 2:Cls
2  Range -10,80,10,-20,20,10
3  79→B:0→C
4  Lbl 1:C-1→C
5  Ran#<0.58⇒C+2→C
6  Plot 80-B,C
7  Dsz B:Goto 1
8  Graph Y=0◢
9  Goto 2
```

THE MATHEMATICS LIBRARY CARTRIDGE FOR THE TI-74 - The program package is very similar to that which was available with the Mathematics module for the CC-40. The two discrepancies with the CC-40 module have been corrected, namely the prompting anomaly in the AU routine (V8N5P18) and the wrong sign for the determinant for some matrices (V10N1P7).

MORE SUBPROGRAMS IN THE STATISTICS MODULE FOR THE CC-40 AND TI-74

V8N6P20 reported that the Statistics cartridge for the CC-40 had subprograms which were not discussed in the manual. One subprogram would input and edit a two-dimensional array in a manner very similar to the MI subprogram in the CC-40 Mathematics cartridge. That capability is also available in the Statistics cartridge for the TI-74, and again there is no mention in the manual. The call is the same as that defined on page 95 of the manual for the Mathematics cartridge for the CC-40:

```
CALL MI(PROMPT$,ARRAY(,),FIRST,LAST ROW,LAST COLUMN,PRINTER)
```

An example will illustrate the use of this call:

```
10  CALL MI("A",A(,),1,2,3,0)
```

The first prompt which will appear is "Enter A(1,1):". As you enter each element of the array the prompts will proceed through "Enter A(1,2):", "Enter A(1,3):", "Enter A(2,1):", etc. If the last variable in the call is a one then each entry is printed together with the accompanying notation. When the last array element has been entered the subprogram enters an edit mode. The prompts and responses for the TI-74 Statistics cartridge are the same as those described for the Mathematics cartridge of the CC-40. The CC-40 Statistics cartridge implementation differs in that it exits the subprogram at the end of an edit of all the input.

Subprogram ME uses the same arguments as subprogram AU but starts execution at the edit routine.

V8N6P20 also noted that there also appeared to be an AU subprogram for input and edit of two one-dimensional arrays in the Statistics cartridge of the CC-40, but the use of the AU call defined on page 87 of the Mathematics cartridge manual yielded a response "Illegal Syntax". After some additional experimentation I have found that a longer call will work for the Statistics cartridges in both the CC-40 and the TI-74:

```
CALL AU(PROMPT1$,PROMPT2$,ARRAY1(),ARRAY2(),FIRST,LAST,
    PRINTER,EXTRA1,EXTRA2)
```

where the EXTRA1 and EXTRA2 elements must be added for use with the Statistics cartridges. An example will illustrate the use of this call:

```
10 N=3
20 CALL UP("Demo of AU in Statistics Module",PN)
30 CALL AU("X","Y",X(),Y(),1,N,PN,0,0)
```

Line 20 is the printer useage subprogram. If the user responds "N" to the prompt "Use Printer?" then PN is set to zero and the computer proceeds to the next line of the program. If the user responds to the prompt "Use Printer?" with "Y" then PN is set to one, and the next prompt asks "Enter File Name:". The user enters the device number. The text inside the quotation marks in the call is printed and the computer proceeds to the next line of the program.

Line 30 is the AU call. This call illustrates that the FIRST, LAST, PRINTER, EXTRA1, and EXTRA2 elements of the call can be constants or variables. The first prompt is "Enter X(1)". When that element has been entered the next prompt is "Enter Y(1)". As each subsequent element is entered the prompt alternates between requests for X or Y elements. When all of the elements of the two arrays have been entered the subprogram enters an editing routine.

More Subprograms for the Statistics Modules - (cont)

The reasons for the elements EXTRA1 and EXTRA2 in the call for the AU subprogram are not clear. Experimentation shows that if EXTRA2 is not zero, then the value for EXTRA1 is added to the value for FIRST to define the subscript for the first prompt for data entry. For example if in line 30 above EXTRA1 is two and EXTRA2 is one, then the first prompt will be "Enter X(3):", the second prompt will be "Enter Y(3):", etc. up through "Enter Y(5):". But if you test the contents of the X and Y arrays you will find that the entered values are in locations with subscripts 1 through 3 not 3 through 5. Thus the EXTRA1/EXTRA2 elements of the call seem to affect the subscripts of the prompt, but not the locations into which the entries are stored! I am at a loss to explain how this is useful. Until we have a better understanding of the effects of the EXTRA1 and EXTRA2 elements of the AU call I recommend that users set them to zero.

Subprogram AE uses the same arguments as subprogram AU but starts execution at the edit routine.

Subprogram RZ checks to see if an input variable is greater than zero. The argument for the call is the same as for the IZ subprogram:

```
CALL RZ(Variable, Flag)
```

If the variable is greater than zero the flag is set to 0. If the variable is less than or equal to zero the flag is set to -1 and the message "VALUE MUST BE > 0" is displayed. However, the message is only displayed for a short period of time. To see the message s user should include a PAUSE statement immediately following the call.

Subprogram IC checks to see that an input value is an integer between two predetermined limits. The call is:

```
CALL IC(Lower, Upper, Variable, Flag)
```

If the variable is not an integer the flag is set to -1 and the message "VALUE MUST BE AN INTEGER" is displayed. If the variable is between the limits the flag is set to zero. If the variable is outside the limits the flag is set to -1 and a message defining which limit was exceeded is displayed. An example will illustrate the use of this call:

```
10 A = -3.5
20 B = 10
30 INPUT N
40 CALL (A,B,N,Q)
50 IF Q=0 THEN PAUSE
60 PRINT "Flag = ";Q:PAUSE
70 GOTO 30
```

Note that the limits do not need to be integers. As with the IZ, RZ and PC subprograms the conditional PAUSE at line 50 permits the user to view the error message. Line 60 allows the user to check the flag response. An input of 5 yields only the message "Flag = 0". An input of yields the message "VALUE MUST BE >=-3.5." followed by the message "Flag = -1". An input of 11 yields the message "VALUE MUST BE <=10." followed by the message "Flag = -1". If the values for A and B are reversed so that the first element of the call is greater than the second element of the call then the flag is set to -1 for all input, and some of the messages are non-sensical. If the two limits are set to the same integer, then only that integer as an input will set the flag to 0.

More Subprograms for the Statistics Modules - (cont)

Subprogram PC checks to see than an input value is greater than zero and less than or equal to one. The argument for the call is the same as for the IZ and RZ subprograms:

```
CALL PC(Variable, Flag)
```

If the variable is inside the defined range then the flag is set to zero. If the variable is outside the defined range then the flag is set to -1 and the message "VALUE MUST BE >0 And <=1" is displayed for a short period of time. Again, a following PAUSE statement should be included.

There is one additional subprogram with a two letter call which is not listed in the Statistics manuals, the TI subprogram. I know a subprogram is there since a call does not yield a "Not Found" response, but I have not yet identified proper arguments for the call. There may also be three letter calls which are not listed in the manuals. Hopefully, we can find a subprogram which will provide for input and edit of a one-dimensional array in a manner similar to the AK subprogram in the Mathematics cartridge.

AREA FINDING - The September 1986 issue of Computer Shopper includes an article entitled "The Universal Area Calculator" by Frank Tymon. A BASIC program was provided for finding the area and perimeter of a polygon defined by its vertices. The method of solution for the area of the polygon is the same as that used in Henrik Klein's polygon program for the TI-58/59 in V8N2P7. Tymon notes that the area calculation will yield a positive answer if the vertices are taken in a counter-clockwise order, and a negative answer if the vertices are taken in a clockwise order. He fails to recognize the capability to traverse the outer polygon and use a "cut" to go from an outer polygon to an inner polygon, followed by a clockwise traverse of the inner polygon to find the area between two polygons. Perhaps he avoided that idea because of the difficulty keeping the "cut" from being included in the perimeter. A CC-40 or TI-74 program which uses the subprograms in the Statistics cartridge to solve for the area is at the right.

```
10 DIM X(31),Y(31)
15 CALL UP("Area Finder",PN)
20 INPUT "Number of Vertices? ";N
25 CALL IC(3,30,N,Q)
30 IF Q=-1 THEN PAUSE:GOTO 20
35 CALL AU("X","Y",X(),Y(),1,N,PN,0,
0)
40 X(N+1)=X(1):Y(N+1)=Y(1)
45 S=0
50 FOR I=1 TO N
55 S=S+X(I)*Y(I+1)-X(I+1)*Y(I)
60 NEXT I
65 PRINT #PN,"Area = ";S/2
70 PAUSE
75 END
```

The prompts provide the instructions for running the program. When making a "cut" it is necessary to enter the transition points between the inner and outer curves both going and returning.

Line 25 uses the IC subprogram to ensure that the number of vertices is at least three and not more than 30. Line 30 illustrates the use of the flag set by the IC routine to branch back to the input if the entered number of vertices was not acceptable.

This program will also operate with the Mathematics cartridge and the CC -40 by deleting line 25 and the last two zeroes in the argument in line 35.

COMBINATORIAL ANALYSIS WITH THE OD, OE, OF HEXCODES ON THE TI-66 - David Douglas

The stimulus of the synthesizing of hexcode OF (15) in the TI-66 in V11N1P24 led to my synthesis of hexcodes OD (13) and OE (14) as others have also probably done by now. I elected to employ these three codes in a program for the number of possible combinations nCr which runs with partitioning 2nd Part 00 (no data registers)! In order to determine the number, type and order of hexcodes required, I first wrote a program referencing data registers 00, 01 and 02 and then substituted codes OD for 00, OE for 01 and OF for 02. When referencing registers in hexcode I found that a CLR must be programmed at steps 005 and 019 (in the program to be synthesized later) before the "1" would properly enter the display register at steps 006 and 020. Also, not knowing how to program OP 3n where "n" is hexcode, I programmed the longer alternative 1 INV SUM OD to decrement register OD beginning with step 020. For the program nine hexcodes must be implanted in this order: OD, OE, OF, OD, OE, OF, OD, OE, and OF. To do that press 2nd Part 64 to set the partitioning to 0.63 (64 registers, no program steps) and then

Keystrokes	Hexcode	In Display
1 EE 50 / EE . 8 +/- 1 +/- x = STO 63 STO 60 STO 57	OD (13)	1. 0-
1 EE 50 / EE . 6 +/- 1 +/- x = STO 62 STO 59 STO 55	OE (14)	1. Or
1 EE 50 / EE . 4 +/- 1 +/- x = STO 61 STO 58 STO 55	OF (15)	1. 0

Press INV EE and recall the contents of the nine data registers in which values were stored above to find 1000 in 63, 60, 57; 10000 in 62, 59, 56; 100000 in 61, 58, 55. Press 2 Op 17 and LRN. SST seventy-two times to see 1's at steps 000, 008, 016, 024, 032, 040, 048, 056, and 064; see the mnemonic ** at steps 007, 015, 023, 031, 039, 047, 055, 063, and 071. There should be zeroes at all other steps. Listing with the PC-200 printer will show the folowing three variations representing the ** mnemonic in the display:

At Step	Printed	Hexcode	At Step	Printed	Hexcode	At Step	Printed	Hexcode
007	007 .	OD (13)	031	031 .	OD (13)	055	055 .	OD (13)
015	01	OE (14)	039	03	OE (14)	063	06	OE (14)
023	023	OF (15)	047	047	OF (15)	071	071	OF (15)

It is interesting to note that the decimal point "." characterizes the OD hexcode, and that a blank mnemonic with the third digit of the step number suppressed characterizes the OE code. With the OF of V11N1P24, this third digit is not suppressed.

Next we must insert the rest of our program for nCr around the implanted hex codes. The procedure is similar to that used in V11N1P24 requiring a lot of SSTing and Deleting. When you finish you should have a program which reads from the display, starting at step 000:

```
000  STO ** R/S STO ** CLR 1 STO ** LBL A RCL ** / RCL ** =
017  PRD ** CLR 1 INV SUM ** DSZ ** A RCL ** R/S RST
```

Now press LRN and RST. Press 2nd Part 00 to partition to see "511" in the display indicating 512 program steps and no data registers. To run the program, press RST, enter n and press R/S, then enter r and press R/S. The program stops with nCr in the display and with the step counter reset to 000 for the start of the next solution. For example, enter 52, press R/S, enter 5, press R/S and in about eight seconds see 2598960 in the display, the number of possible combinations of 52 playing cards taken 5 at a time (the possible poker hands), all of this accomplished with a partitioning that indicates that no data registers are available. (But see further discussion on page 26).

Editor's Note: To save space I deleted David's step-by-step instructions for inserting the program sequence around the implanted hex codes. If you have difficulty, send a SASE and I will send the instructions.

MANY DIGITS OF LN(2) AND LN(3) IN BASIC

One of the programs in Robert Prins' treatise on extended precision calculations for the TI-59 (see page 25) can calculate natural logarithms up to 1188 digits. The problem was to find an independent listing to verify that the program was correct. I was able to find fifty digits of ln(2), ln(3) and ln(10) on page A-1 of the CRC Handbook of Chemistry and Physics, but was unable to find any longer listing. Robert and I decided to independently calculate 1200 digits using other programs and compare our results. My program used the well known relationship

$$\ln\frac{1+x}{1-x} = 2\left[x + \frac{x^3}{3} + \frac{x^5}{5} + \frac{x^7}{7} + \ldots\right]$$

which is also the basis for Robert's program for the TI-59. For x = 1/3 that series reduces to

$$\ln(2) = 2\left[\frac{1}{3} + \frac{1}{3\cdot 3^3} + \frac{1}{5\cdot 3^5} + \frac{1}{7\cdot 3^7} + \frac{1}{9\cdot 3^9}\ \ldots\right]$$

I used a brute force approach since memory limitations weren't a problem with my CC-40. The listing for the program that I used on the CC-40 is at the right.

```
100 DIM A(120),B(120),C(120)
110 IMAGE ##########
120 CALL UP("Logarithm of 2",Z)
130 S=1.E+10
140 INPUT "Number of 10 Digit Blocks
 ? ":N
150 FOR I=1 TO N
160 A(I)=6666666666
170 NEXT I
300 I=1
400 R=0
410 FOR J=1 TO N
420 B(J)=A(J)+R*S
430 R=B(J)-I*INT(B(J)/I)
440 B(J)=INT(B(J)/I)
450 NEXT J
500 M=(I+1)/4
510 R=2*(M-INT(M))
520 FOR J=N TO 1 STEP -1
530 C(J)=C(J)+B(J)+R
540 R=INT(C(J)/S)
550 C(J)=C(J)-R*S
560 NEXT J
600 R=0
610 FOR J=1 TO N
620 T=INT((A(J)+R*S)/9)
630 R=A(J)+R*S-9*T
640 A(J)=T
650 NEXT J
700 IF B(N)<>0 THEN I=I+2:GOTO 400
800 PRINT #Z,10*N;" Digits"
810 PRINT #Z
820 FOR J=1 TO N
830 PRINT #Z,USING 110,C(J)
840 IF Z=0 THEN PAUSE
850 NEXT J
900 IF Z=1 THEN CLOSE #1
999 END
```

Line 100 sets up 3 extended precision registers (arrays) where ten digits to the left of the decimal point are maintained in each of the 120 elements of each array for a total of 1200 digits. The digits to the right of the decimal point are used to carry from one element to the next.

Line 120 assumes the user has one of the cartridges installed.

Line 140 allows the user to select how many digits to calculate. Using only the number of digits needed saves calculation time.

Lines 150-170 load the A array with the value 2/3, the first term of the series. Lines 400-450 divides the A array by 2n-1 and accumulates the result in the B array.

Lines 500-510 provide automatic compensation for truncation effects by adding 1 at the least significant digit every other cycle, for an average rate of 0.5 of the least significant digit per cycle which will be approximately correct if the truncation errors are uniformly distributed. Without this correction a series with all positive terms will generate an error equal to one-half of the number of terms times the least significant digit.

Lines 520-560 complete the addition of array B to array C to accumulate the sum.

Many Digits of Ln(2) and Ln(3) - (cont)

Lines 600-650 divide the contents of the A array by nine (3 squared) to prepare for the next iteration.

Line 700 keeps the routine iterating until there are no more corrections to array C. This is easier than trying to figure out how many cycles to use to get a given number of digits.

Lines 800-840 provide printout of the result once the calculations are complete.

To calculate ln(3) you only need to note that x should be 1/2 in the beginning formula. In the program you change line 160 to 9999999999 (ten nines). You do not start with a 1 and ten zeroes in A(1) and zeroes in the rest of the elements of A as you might think to avoid termination after the first term by line 700.

The listings at the right are for 1000 digits of ln(2) and ln(3). Robert and I agree on ln(2) but he has not yet completed ln(3). The first 50 digits of ln(3) agree with the CRC values.

EXTENDED PRECISION PROGRAMS FOR THE TI-59

Member Robert Prins has put together an extensive treatment of extended precision programs for the TI-59. Many of the programs which have appeared in TI PPC Notes, Programbiten, etc., over the years are included, often in optimized form. There are ten chapters plus five appendices for a total of 98 pages. Each chapter includes a short description of the program and its origin, a description of the method used, user instructions and a program listing. The programs provide for extended precision calculation of factorials, e, pi, natural logarithms, and roots. In some cases programs are provided for the TI-58 and TI-66. The appendices discuss methods for speeding execution, tricks to save program steps for the TI-66, some TI-57 extended precision programs, and some representative results.

This is a must document for those who are seriously interested in extended precision programming. Robert offers the compilation for FL 40 by INTERNATIONAL postal money order for surface mail. For air mail service, the price is FL 48. If you wish you may send the equivalent in cash in U.S. dollars. Write to Robert Prins, Alfred Nobellaan 112, 3731 DX De Bilt, Netherlands.

Logarithm of 2 — 1000 Digits	Logarithm of 3 — 1000 Digits
6931471805	986122886
5994530941	6810969139
7232121458	5245236922
1765680755	5257046474
13436025	9055782274
5254120680	9451734694
94933936	3336374942
2196969471	9321860896
5605863326	6873615754
9964186875	8137320887
4200148102	8797002906
570685733	5957865742
6855202357	3680042259
5813055703	3051982105
2670751635	2801870767
759619307	2774106031
2757082837	6270918336
1435190307	1367179373
386238916	6988443609
7347112335	5990374257
115364497	316795911
9552391204	5211455919
7517268157	1775807134
4932065155	7054940166
5247341395	7755802222
2588295045	317025294
3007095326	6897560690
3666426541	1005215056
423915781	4206613803
4952043740	6317373298
4303855008	5777823669
194417064	9105479213
1671518644	1818149022
7128399681	301838236
7178454695	3012224865
7026271631	2748198225
645461502	9910974524
5720740248	9089645805
1637773389	3407808845
6385506952	9050857484
6066834113	4411901885
7273873722	7087047494
9289564935	8070796130
4702576265	8582941100
2098859693	2100121184
2019650585	14098255
5476470330	1439194876
6793654432	8893079849
5476327449	4302255731
5125040606	5353290853
9438147104	4529525145
6899465062	9213670494
2016772042	6859325627
4524529612	9441655694
6879465461	1578272310
9316517468	3551088061
1392672504	211846989
1038025462	439943063
5965686914	1302552857
4192871608	3646088282
2938031727	4988130822
1436778265	8006341439
4877566485	1070089325
856740776	1450437510
4845146443	8044510275
9940461422	6193497398
6031930967	2110941585
3540257444	7405353617
6070308096	6890097512
850474866	2233797730
3852313818	9090877543
1676751438	6479513571
6674766478	2982177817
9088143714	5812421223
1985494231	5140581010
5199735488	3272405586
375165861	9372495649
2753529166	1918524290
1000710535	796084234
5824987941	6470693772
4729509293	3725205508
1138971559	2032078333
9820565439	9280558928
2871700072	5314087309
1808576102	5132000458
5236889213	3091843974
2449713893	9082223032
2037843935	5705407533
3088774825	3118230190
9701715591	4927525759
708823683	9132217851
6275898425	3533902374
8918535302	8290433950
4363421436	2540874245
7061189236	8249340068
7891923723	6012188143
1467232172	6520565429
534016492	5427070105
5687274778	547779542
2344535353	2933973343

TI's DEALER PARTS FACILITY - The mailing envelope for my replacement TI-55III (see page 12) contained an order form for the dealer parts activity at TI. Some representative prices include:

1015767-3950	BP1A battery pack for the TI-58/59	$12.95
1220900-3950	BP2	7.00
1014791-3950	BP3	4.50
1014421-3950	BP6 battery pack for the SR51-II	12.95
1016265-1	BP7 battery pack for the TI55/57/MBA	10.95
1016002-1	BP8 battery pack for the SR40/TI30	9.95
1030265-1	PC-100 Paper (3 rolls)	10.00
1052802-101	PC-200 Paper (3 rolls)	2.95
	Service Manuals	10.01

Generally, those prices are not as good as are available from Elek-tek, but not all of the items are available elsewhere. The order form lists a shipping handling charge of $2.50 for orders up to $30.00, and $3.00 for orders over $30.00. Write to Texas Instruments, P.O. Box 53 Attn: Dealer Parts, Lubbock TX 79408 or call 1-806-741-3090.

ON TI-66 HEXCODES AND OPERATION WITH NO DATA REGISTERS - P. Hanson

See page 23. Of course, there must be some data registers somewhere. We might have hoped that we had found some new registers such as were found with the SR-52; however, a quick test shows that is not the case. If after completing the sample problem on page 23 you press RCL 15 you get "...E.r.r.o.r.." in the display. Then press CLR 2 OP 17 RCL 15 and see the answer 2598960. Press RCL 13 and see 47. Press RCL 14 and see zero, the result of Dszing data register OE (014). We conclude that when the program is run in the 2nd Part 0 mode the decimal equivalent registers are actually used. The "quirk" is that with hexcodes you can access data registers which would otherwise not be available due to partitioning. This should allow us to write self-modifying programs without changing the partitioning to change the modify the program.

TABLE OF CONTENTS

Magnetic card service is available for the programs in this issue, and for the 1983, 1984, and 1985 issues, for the price of one dollar per card plus a stamped and self-addressed envelope.

CCL-144 Cleaning Strips are two dollars each. No checks please.