DISPLAYING COMBINATIONS - Larry Leeds wrote that he had a problem which required the display (or printout) of all available combinations of n things taken r at time. That is different from the standard calculation of the number of combinations (nCr). In response I noted that the solution for the case where r = 3 was implicit in the smallest circle program (see lines 345-355 and 735-760 on V13N1P29). I used those steps as the starting point for the short TI=74 program at the right. A sample output appears below the program.

```
100 INPUT "Use PC-324? Y
/N ";Z$
110 IF Z$="Y"OR Z$="y"TH
EN PN=1 ELSE 130
120 OPEN #1,"12",OUTPUT
130 PRINT #PN
140 PRINT #PN,"nC3 Combi
nations":PAUSE 1
150 INPUT "n = ";N
160 PRINT #PN,"n = ";N:P
AUSE 1
200 FOR I=N TO 3 STEP -1
210 FOR J=(I-1)TO 2 STEP
 -1
220 FOR K=(J-1)TO 1 STEP
 -1
230 PRINT #PN,I;J;K
240 IF PN=0 THEN PAUSE
250 NEXT K
260 NEXT J
270 NEXT I
999 END
```

```
nC3 Combinations
n =  5
 5  4  3
 5  4  2
 5  4  1
 5  3  2
 5  3  1
 5  2  1
 4  3  2
 4  3  1
 4  2  1
 3  2  1
```

In a few days Larry responded with a BASIC program which will handle values of r from 2 through 9. I converted his program for the TI-74. I also incorporated the prompt for the use of the Printer 80 (see V14N3P14) in line 50. The program listing as obtained from the Printer 80 appears on page 9.

Sample printouts with various printers which can be interfaced with the TI-74 appear on page 10. For those printouts the PC-324 was connected to the TI-74 through the built-in cable, the HX-1000 was connected to the PC-324 through the adapter cable defined by Maurice Swinnen in V12N3p13, and the Printer 80 was connected to the HX-1000 through a standard hex bus cable. An appropriate power supply was connected to each printer yielding a veritable rat's nest of cabling on my kitchen table.

On the surface Larry's program appears to be a "brute force" implementation which could be substantially shortened with the use of subscripts, subroutines, ON-GOTO's and the like. So far I haven't had much success in optimization, primarily because the BASIC implementations at my disposal (Microsoft BASIC in the Model 100 and TI BASIC in the CC-40 and TI-74) do not support the use of subscripted variables as the index in FOR-NEXT statements. Members are invited to submit improvements on Larry's program where the desired result (at least in my view) would be a program which could handle any r within the range of a DIM statement.

Displaying Combinations - (cont)

Program Listing on the Printer 80

```
10 Y$="nCr Combinations":PRINT Y$:PAUSE 1
15 REM 1 April 1991
25 INPUT "Use printer? Y/N ";Z$
30 IF Z$="Y"OR Z$="y"THEN PN=1 ELSE 100
35 PRINT "Device Numbers:":PAUSE 1
40 PRINT "For the HX-1000 enter 10":PAUSE 1
45 PRINT "For the PC-324 enter 12":PAUSE 1
50 PRINT "For the Printer 80 enter 16":PAUSE 1
55 INPUT "Enter device number ";D$
60 OPEN #1,D$,OUTPUT
65 IF D$="10"THEN PRINT #1,CHR$(18)
75 PRINT #1:PRINT #1,Y$:PRINT #1
100 INPUT "n = ";N
105 INPUT "r (r<n; 1<r<10) = ";R
110 IF R<2 OR R>9 OR R>N-1 THEN PRINT "r is out of range" ELSE 120
115 PAUSE:GOTO 100
120 P$=STR$(N)&"C"&STR$(R):PRINT #PN,P$
125 IF PN=0 THEN PAUSE ELSE PRINT #1
130 M=0:PAUSE ALL
135 FOR A=N TO R STEP -1
140 FOR B=A-1 TO R-1 STEP -1
145 IF R=2 THEN M=M+1:PRINT #PN,A;B:GOTO 290
150 FOR C=B-1 TO R-2 STEP -1
155 IF R=3 THEN M=M+1:PRINT #PN,A;B;C:GOTO 280
160 FOR D=C-1 TO R-3 STEP -1
165 IF R=4 THEN M=M+1:PRINT #PN,A;B;C;D:GOTO 270
170 FOR E=D-1 TO R-4 STEP -1
175 IF R=5 THEN M=M+1:PRINT #PN,A;B;C;D;E:GOTO 260
180 FOR F=E-1 TO R-5 STEP -1
185 IF R=6 THEN M=M+1:PRINT #PN,A;B;C;D;E;F:GOTO 250
190 FOR G=F-1 TO R-6 STEP -1
195 IF R=7 THEN M=M+1:PRINT #PN,A;B;C;D;E;F;G:GOTO 240
200 FOR H=G-1 TO R-7 STEP -1
205 IF R=8 THEN M=M+1:PRINT #PN,A;B;C;D;E;F;G;H:GOTO 230
210 FOR I=H-1 TO R-8 STEP -1
215 PRINT #PN,A;B;C;D;E;F;G;H;I:M=M+1
220 NEXT I
230 NEXT H
240 NEXT G
250 NEXT F
260 NEXT E
270 NEXT D
280 NEXT C
290 NEXT B
300 NEXT A
310 PAUSE 0:PRINT #PN
320 PRINT #PN,P$&"= ";M:IF PN=0 THEN PAUSE
330 PRINT #PN:GOTO 100
999 END
```

<u>"HANG-UP" OF THE TI-95</u> - Peter Messer called to ask if I had ever experienced a "hang-up" condition when using the TI-95 with the PC-324 printer. I replied that I remembered having the problem once when operating with a PC-324 with dead batteries but had difficulty duplicating the condition. Peter had similar difficulty in duplicating the condition, and seemed to get different results with different calculators.

The "hang-up" can be cleared by pressing the RESET button. Fortunately, as indicated on page A-2 of the *TI-95 User's Guide* the use of the RESET button does NOT clear data registers or program registers.

<u>BATTERY LIFE IN THE TI-81</u> - A few weeks after the purchase of my first TI-81 I found that I needed to change the batteries. I thought that might be a result of some unusually heavy useage as I became familiar with the device. I loaned that unit to my daughter in college for the spring semester. Less than two months into the semester she called to report that the TI-81 was "acting up". I coached her in the use of the display contrast indicator as an indicator of battery status from page 1-5 of the *TI-81 Graphics Calculator Guidebook*. The indication was that a battery change was required. I also needed a battery change on my second TI-81 after only a few weeks of use. The TI-81 uses four AAA batteries and I have always used the alkaline versions. My fx-7000G's which use three lithium CR2032C batteries have operated for a year or more on a set. Have others experienced unexpectedly short battery life with the TI-81?

<u>BATTERY USEAGE IN THE PC-324</u> - My PC-324 is also "battery eater" even though I seldom use it without an AC9201 attached. Has anyone else had this experience, or do I have a defective unit?

<u>CC-40 HARDWARE AND SOFTWARE WANTED</u> - write to Kelvin Cane,
3111 Artaban Place
Baltimore MD 21216

<u>CC-40 HARDWARE WANTED</u> - If you have an Editor/Assembler cartridge, a video interface or a Wafertape drive in working order write to M. J. M. Wright, 45 Centerville Drive, Salem, NH 03079.

<u>TI-74 EQUIPMENT FOR SALE</u>

TI-74	BASICCALC with manuals	$ 80.00
PC-324	Portable Printer with Paper-holder bracket	70.00
AC-9201	Adapter	10.00
PA-201	AC Interface	5.00
CI-7	Cassette Interface	20.00
CTR-73	Tape Recorder - Realistic	35.00
	TI-74 Learn BASIC Guidebook	5.00

A substantial reduction in price will be offered on a package deal. This equipment was all purchased new in October 1989 and used in one project and not used since. Write to Carl Rabe, P.O. Box 2941, Santa Maria, CA 93457-2941.

A HISTORICAL NOTE - from BITS, BYTES & PIXELS, Volume 6, Number 9

NEVER RELEASED OFFICIAL TI PERIPHERALS: THE WAFERTAPE DIGITAL TAPE DRIVE

a hands on description by Charles Good
Lima Ohio User Group

The WAFERTAPE DIGITAL TAPE DRIVE was supposed to be a step up from cassette data storage. The device is totally under the control of the computer (no manual rewinding or keeping track of cassette counter numbers). A directory on the tape allows the computer to automatically advance the tape to the beginning of any desired file. Recording data digitally (as 1 or 0, on or off) rather than as part of a continuous spectrum of sound frequencies as is done on a regular sound cassette recorder such as T.I.'s Data Recorder, was supposed to be a more reliable way of recording and retrieving data. Had this device been released in 1983 after its debut in January of that year, it and the also never released HexBus interface would have formed an inexpensive mass storage upgrade (compared to using a cassette recorder) for the 99/4A at a time when a new full PE box (SSSD drive, 32K, disk controller) cost anywhere from $550-$1200. The last 99/4A catalog published by TI in the fall of 1983 lists the Wafertape drive for $139.95 and the HexBus interface for $59.95.

The wafertape drive was shown at the Consumer Electronics Show in January 1983 together with the first showing of the CC40. It was to be the major mass storage device for the CC40, and is described and pictured in the user guides that come with most CC40 software and peripherales. Unfortunately these user guides now also come with an addendum sheet that states "The Wafertape Digital Tape Drive is not available." The non release of the Wafertape Drive left the CC40 totally without a mass storage device until the 1986/1987 introduction of the Mechatronic HexBus Quickdisk drive. This lack of a mass storage device probably killed most consumer interest in the CC40.

I recently purchased for $100 a working Wafertape Drive, serial number 0000007, ATA3883. It is my understanding that the ATA number is a date code, indicating in this case manufacture in the 38th week of 1983. Most 99/4A hardware and software modules have an ATA number. If my understanding of ATA numbers is correct, my Wafertape Drive was not one of those shown at the January 1983 exhibit. I personally know of one other working Wafertape drive, serial number 0000095. I have been told that there are about 10 or 11 working Wafertape Drives in private hands and others that don't work.

The Wafertape Drive is comparable in size to other TI HexBus peripherals, measuring about 11.5cm wide, 14.5cm deep and 3.5cm tall. It is designed to be stacked with the HexBus RS232, modem, and Printer/Plotter. It is battery powered with 4 AA cells. You can also use an AC adapter. The front has an on/off switch and a slot for inserting the wafertape. On the back are two HexBus ports, an AC adapter jack, and a rotary switch for setting the device number. The switch on mine has positions 0-9. However, only positions 0-7 work, corresponding to devices 1-8. By accessing each HexBus peripheral individually by number, a single CC40 could control up to 8 Wafertape Drives, if one could somehow gather that many working drives together in one place.

Wafertapes come in a cartridge measuring 68x40x5 mm, about as big as the miniature cassettes sold these days for small tape recorders. The top is clear plastic and the bottom is black plastic. T.I. calls these cartridges "wafers". They fit easily and snugly into the slot in the front of the Wafertape Drive. Inside the wafer a dark colored magnetic tape 1.7mm wide (very thin) is wound in a continuous loop, in the same manner as the tape of an "8 track" music cartridge. The tape moves only in one direction and its ends are attached to each other with a piece of reflective silver tape. T.I.'s last 99/4A catalog lists 50 foot ($7.95), 25 foot ($6.95), and 10 foot ($5.95) wafertapes. Five foot wafertapes are also mentioned in the Wafertape User's Guide. Long tapes store more, but it takes longer to find the beginning of a specific file. Official T.I. wafers have a little sliding panel that covers the exposed part of the tape at the edge of the wafer. This slides open as the wafer is inserted into the slot on the Wafertape Drive. I have one such T.I. wafer and I also have some wafers without a T.I. label and without the sliding panel. I wonder who made these "generic" wafers? I do know that other wafertape drives were planned or actually sold for other devices. A wafertape drive for the Tandy 100 laptop computer is described in the March 1984 issue of Creative Computing. Also, see the comments below from Tony McGovern about another wafertape device. Maybe my "generic" wafertapes were not made specifically for the TI Wafertape Drive.

The method of write protecting wafers is unusual. When I first got my Wafertape Drive I thought it was defective because I always got a "write protected" error message whenever I tried to initialize a wafer. I could find nothing resembling a write protect tab on my wafers, and looking inside the Wafertape Drive slot revealed no evidence of a mechanical pin associated with the "remove the tab and it is write protected" system of protection typically found on most audio and video tape cartridges. It turns out that you have to put a "write enable" paper sticker at a specific location on the top of the wafer in order to write to the wafer. Anything white will do. This "write enable" piece of paper is OPTICALLY sensed from above by the Wafertape Drive. In the absence of the sticker, the optical sensor sees through the transparent wafer upper surface and does not get a reflection off of the black wafer bottom. A second optical sensor detects the silver end/beginning of tape marker. This marker is the only reference point the wafertape drive has to tell the relative position of everything else on the tape. On "8 track" continuous loop music cartridges, the end/beginning of tape marker is detected electronically. This silver marker on a wafertape is detected through a

A Historical Note - (cont)

window in the center of the wafer's transparent upper surface. If the "write enable" sticker you are using is too big and covers this window, then the wafer will not be usable as I finally figured out after lots of frustration.

When you initialize a wafer with the FORMAT command of the CC40, the tape is advanced until the marker is optically detected, and then the Wafertape Drive prepares a new tape directory area at the beginning of the tape. The entire tape is not magnetically encoded with FORMAT, just the directory area. Software designed to directly read the contents of a wafertape directory reveals that if a previously used wafer is reFORMATted, the old file names are retained in the directory area, but all file lengths are set to zero. The time required for a FORMAT depends on how close the tape is to its beginning when FORMAT begins. Even a 50 foot tape formats very rapidly if it is already almost at its beginning.

A directory has room for 16 files irrespective of tape length. Of course short tapes may not have enough room for 16 files if they are of significant length. Files are written to the tape sequentially, with the directory keeping track of the file name, number (0-15), and length. Apparently the Wafertape Drive locates specific files by file number, counting the End-Of-File indicators that pass by the read/write head as the tape is advanced to the start of the desired file. Only the last file can be overwritten by another file of the same name. If you write a file with the same name as a file already on the wafer that is not the last file on the wafer, the old file's directory reference is deleted and the new file of the same name is written to fresh space after all the other files currently on the tape. (I hope you understood that.) File types supported include PROGRAM, and INTERNAL or DISPLAY data files. RELATIVE files are not supported, and you can't open data files as APPEND. Although this is not made clear in available documentation, I think that when data files are read from wafertape, the entire file is read into computer memory for manipulation by the controlling program and then later if necessary written back to wafertape. This limits the size of data files. Only one wafertape file can be opened at a time.

To compare the speeds of the Quickdisk and Wafertape drives, I timed the SAVE and OLD of a 15300 byte text file from Memo Processor using a newly FORMATed wafer and disk. Wafertape: SAVE, 4 min 25 sec; OLD; 3 min, 10 sec. Quickdisk: SAVE, 2 min 15 sec; OLD, only 38 seconds.

Why wasn't the Wafertape Digital Tape Drive ever officially released? It just did not meet T.I.'s standards for reliability. It does not work well on battery power. Even with four newly installed, fresh alkaline AA batteries, you almost always get an I/O error 25 (low batteries in peripheral) when you try to load something, and you often get the same error when you try to SAVE while on battery power. The Wafertape Drive only works with much reliability when operated with the AC adapter. Apparently the speed at which the tape crosses the Wafertape Drive's read/write head is critical, and variations in this speed are not tolerated. With any battery, continuous power drain results in a voltage decrease compared to the initial voltage put out by the battery. Such a voltage decrease slows down the drive motor. Also, as the Wafertape Drive operates it turns itself on and off several times as it loads or stores data. Starting an electric motor requires an immediate surge of extra current compared to the current needed to keep the motor operating at constant speed once it has started. It is possible that the Wafertape Drive's AA batteries are not able to maintain constant voltage with all the required on/off cycles. In addition to the battery problem, it is sometimes possible to write data beyond the end of the wafertape and wipe out the directory on the other side of the reflective end/beginning of tape marker. This renders all files on the tape useless. I have managed to overwrite the end of a wafertape on two occasions. At other times, when I deliberately tried to do this, the CC40 would not let me write past the end of a wafertape. I sometimes get I/O error 6 (device error, try again) with the Wafertape drive for no reason I can determine. Sometimes trying again doesn't work. I never get these error messages when using my Quickdisk drive. Finally, I suspect that wafertapes are not as durable as disks. The tape is very tiny and is subject to a lot of physical movement and twisting as it moves within the wafer. I suspect that with time the tape may break. I know for example that I have had a higher percentage of my "8 track" music tapes break compared to my reel to reel cassette music tapes. Let me quote from a letter received recently from Tony McGovern of Australia, senior author of FUNNELWEB. "Wafer-tapes were always a disaster area! I think they appeared in one of Sinclair's UK machines. The other place they appeared, also in the UK, was in an abomination produced by ICL sold here, a computer phone -a combination of low end PC with modem/phone all built in- but no disk drive, only the wretched unreliable wafertape. Telecom Australia probably has a warehouse full of these things that they would rather not be reminded of. They had a great marketing campaign to sell these several years ago and no one wanted the turkeys."

I do use my Waertape drive. It isn't that unreliable. I have a little briefcase in which I can keep my CC40, the Wafertape Drive, my HexBus printer/plotter, a power strip, and all the necessary AC adapters all pluged in and ready to go. I can open the briefcase and plug in the power strip and use the peripherals as they sit in the briefcase. I can also use the printer/plotter and CC40 while still in the briefcase using battery power, but I have given up trying to use the Wafertape Drive with battery power. I keep my HexBus RS232 and Quickdisk drive pluged in (these two devices both REQUIRE AC power) next to my 99/4A. If possible, anything saved to wafertape ALSO eventually gets save to disk with the Quickdisk drive, which I find to be quite reliable.

<u>The</u> <u>TI-74S</u> - V14N1P5 described a new hand-held calculator from TI which is similar to the TI-74 but has a simplified keyboard and the capability to handle ROM modules with larger memory. We stated that the device was called the TI-74B. Tom Ferrio of TI wrote that the device is actually the TI-74S.

<u>AN INADEQUACY IN THE LINEAR EQUATION SOLUTION FROM V8N6P20</u> - This BASIC routine was extracted from a least squares polynomial solution. It was used successfully in V8N6P18-19 to solve the 7x7 sub-Hilbert as a comparison of the capability of various machines. It was also used without any problems in several least squares solutions for the TI-74 such as the regression with user defined functions on V12N1P14, the interative regression on V12N2P21, the double exponential evaluation of data program on V13N3P19, and the regression with two or three variables on V13N4P22. But, page 18 of this issue reports the inability of the routine to solve linear equations when a diagonal element of the matrix is zero. The problem is caused by the divisions by zero in the solution; for example, at lines 335 and 340 in the listing on page 19. The divides were not a problem when the routine was used in least squares solutions because there the diagonal elements are by definition greater than zero.

So, while the "linear equation" solution routine from V8N6P20 should be useable in all cases when coupled with a least squares solution it will not gnerally be useable as a linear equations solver. If some elements of the input matrix can be zero then the user will be better served by using the linear equation solver from V12N4P13.

<u>GENERATING SBR COMMANDS WITH HEXADECIMAL ADDRESSES</u> - V14N1P24 reported that Robert Prins had found a way to synthesize hex code addresses in user memory. I delayed publication of the details at that time since I had generated the hex code addresses but was unable to get the resulting program to run. With Robert's help I subsequently realized that the running problem was independent of the hex code addresses.

Suppose that we wish to replace the sequence 24637 STO A 'MTH' RUN SBR IND A with the more direct 'MTH' RUN SBR 603D, where either routine provides entry at the NEW option of the INV/LINEAR SYS menu in the MATRIX ALGEBRA portion of the Mathematics Library of the TI-95. First, you must have set the system mode option of the FUNC key. Then enter LEARN mode and proceed to the place you wish to generate the SBR 603D code. Enter the code

```
DFN 1 2 3 4 5 RCB FA60 '='
```

where there is no Fx after the DFN, just the five digits, and the = sign was entered in the ALPHA mode. Use the back arrow key to backstep until the flashing cursor is over the 1. Press 2nd F:CLR F1, and backstep once more. Press 2nd DEL to delete the DFN F1:???@?? leaving the code SBR 603D.

Some insight as to what is going on can be obtained by noting that the Program Code Table on pages C-4 and C-5 of the *TI-95 Programming* Guide shows that FA is the code for the function SBR and 3D is the code for the character =. Now, suppose that you would like to generate the SBR 70A3 command, which is the entry point for the phi subprogram in the Mathematics module. Devise your solution and refer to page 6 of this issue for the answer.

The technique is in direct contradiction of the statement by TI that "... the SBR command can not take a hexadecimal value with a letter value (A-F) in it ..." (see V14N1P5).

FLATNESS - P. Hanson. Don Laughery proposed this problem. The idea is to use a best fit plane as a measure of the "flatness" of a surface. The input would be a set of X, Y, and Z points relative to a reference plane. The output would be the deviations from the best fit plane, where the maximum deviations would be of primary interest. Pages 221-222 of Spiegel's *STATISTICS* in the Schaum's Outline Series states:

> "... For example, there may be a relationship between the three variables X, Y and Z which can be described by the equation
>
> $$Z = a_1 + a_2X + a_3Y$$
>
> which is called a linear equation in the variables X, Y and Z.
>
> In a three dimensional rectangular coordinate system this equation represents a plane and the actual sample points (X_1, Y_1, Z_1), (X_2, Y_2, Z_2), ... (X_n, Y_n, Z_n) may "scatter" not too far from this plane which we can call an approximating plane.
>
> By extension of the method of least squares, we can speak of a least squares plane approximating the data. If we are estimating Z from given values of X and Y, this would be called a regression plane of Z on X and Y. ..."

This is, of course, exactly the sort of problem that was solved by the programs in V12N3P19 and V13N3P19. For this special application we made several changes relative to those programs:

* The variable assignment was changed so that X and Y were the independent variables and Z was the dependent variable to agree with user's conventions.

* A routine was added to find the maximum deviations from the best fit plane (steps 605-610).

* A routine was added to permit editing of previously entered data (steps 700-740).

A listing of the program for the TI-74 appears on the opposite page. Two printouts for a sample problem appear at the right. The upper printout is for a problem where one of the values was entered incorrectly. The lower printout is for the same problem with correct input. Note the reduced residuals and standard error.

Interpretations:

The A1, A2 and A3 values in the output are the coefficients of the equation of the plane. A1 is the z-axis intercept, obtained by setting x and y to zero. Similarly, -A1/A2 is the x axis intercept, and -A1/A3 is the y axis intercept.

The residuals from the best fit plane are calculated along the Z axis, not normal to the plane.

```
X1 =          1
Y1 =         -9.99
Z1 =          40.16

X2 =          2
Y2 =         -4.98
Z2 =          28.74

X3 =          3
Y3 =          0
Z3 =          17.34

X4 =          4
Y4 =          5.01
Z4 =          5.9

X5 =          5
Y5 =          9.98
Z5 =         -5.53

X6 =          6
Y6 =          15
Z6 =         -16.95

X7 =          7
Y7 =          19.98
Z7 =         -28.37

A1 =  50.8724644
A2 = -11.18359307
A3 = -.0479041414

d1 = -.0074337037
d2 = -.0038408901
d3 =  .0183147992
d4 =  .0019076128
d5 = -.0064157392
d6 = -.0023438842
d7 = -.0001881948

dmax =  .0183147992

dmin = -.0074337037

Mean = -7.142857E-13

S.E. =  .010674434
```

```
X1 =  1
Y1 = -9.99
Z1 =  40.16

X2 =  2
Y2 = -4.98
Z2 =  28.74

X3 =  3
Y3 =  0
Z3 =  17.32

X4 =  4
Y4 =  5.01
Z4 =  5.9

X5 =  5
Y5 =  9.98
Z5 = -5.53

X6 =  6
Y6 =  15
Z6 = -16.95

X7 =  7
Y7 =  19.98
Z7 = -28.37

A1 =  53.55750988
A2 = -12.08011949
A3 =  .1317364727

d1 = -.0013430286
d2 = -.0012232672
d3 =  .0028485882
d4 =  .0029683496
d5 = -.0016424302
d6 = -.0028400336
d7 =  .0012318219

dmax =  .0029683496

dmin = -.0028400336

Mean =  4.285714E-13

S.E. =  .0028507199
```

Flatness - (cont)

```
10 REM Least Squares Plane with
20 REM Identification of the
30 REM Maximum Deviations from the Plane
40 REM 10 June 1990
100 DIM A(8,8),B(8),F(8),X(50),Y(50),Z(50)
105 INPUT "Use Printer <Y/N>? ";A$
110 IF A$="Y"OR A$="y"THEN PN=1 ELSE 130
115 INPUT "Device Code ? ";P$
120 OPEN #1,P$,OUTPUT
130 INPUT "Number of Data Points? ";K
140 FOR I=1 TO K
145 P$=STR$(I)&" = "
150 A$="X"&P$:INPUT A$;X(I)
160 IF PN<>0 THEN PRINT #PN,A$,X(I)
170 A$="Y"&P$:INPUT A$;Y(I)
180 IF PN<>0 THEN PRINT #PN,A$,Y(I)
181 A$="Z"&P$:INPUT A$;Z(I)
183 IF PN<>0 THEN PRINT #PN,A$,Z(I)
185 PRINT #PN:IF E$<>""THEN 700
190 NEXT I
200 N=3:PRINT "Solving"
210 FOR I=1 TO N:FOR J=1 TO N
220 A(I,J)=0:NEXT J
230 B(I)=0:F(I)=0:NEXT I
240 FOR L=1 TO K
250 GOSUB 1000
300 FOR I=1 TO N:FOR J=1 TO N
305 A(I,J)=A(I,J)+F(I)*F(J):NEXT J
310 B(I)=B(I)+F(I)*F(0):NEXT I
315 NEXT L
320 FOR L=1 TO N
325 P=A(L,L)
330 FOR J=L TO N
335 A(L,J)=A(L,J)/P:NEXT J
340 B(L)=B(L)/P
345 FOR I=1 TO N
350 IF I=L THEN 375
355 G=A(I,L)
360 FOR J=L TO N
365 A(I,J)=A(I,J)-G*A(L,J):NEXT J
370 B(I)=B(I)-G*B(L)
375 NEXT I
380 NEXT L
400 FOR I=1 TO N
410 X$="A"&STR$(I)&" = "
420 PRINT #PN,X$;B(I)
430 IF PN=0 THEN PAUSE
440 NEXT I
450 PRINT #PN
500 INPUT "Display Residuals <Y/N>? ";A$
510 S1=0:S2=0:MAX=0:MIN=0
520 FOR L=1 TO K
530 GOSUB 1000
540 ZF=0:FOR J=1 TO N
550 ZF=ZF+B(J)*F(J):NEXT J
560 D=F(0)-ZF
570 IF A$="N"OR A$="n"THEN 605
580 P$="d"&STR$(L)&" = "
590 PRINT #PN,P$;D
600 IF PN=0 THEN PAUSE
605 IF D>MAX THEN MAX=D
610 IF D<MIN THEN MIN=D
615 S1=S1+D:S2=S2+D*D:NEXT L
620 PRINT #PN
625 PRINT #PN,"dmax = ";MAX
630 IF PN=0 THEN PAUSE
635 PRINT #PN
640 PRINT #PN,"dmin = ";MIN
645 IF PN=0 THEN PAUSE
650 PRINT #PN
655 PRINT #PN,"Mean = ";S1/K
660 IF PN=0 THEN PAUSE
665 PRINT #PN
670 PRINT #PN,"S.E. = ";SQR(S2/(K-N))
675 IF PN=0 THEN PAUSE
680 PRINT #PN
700 INPUT "Edit Input Data <Y/N>? ";E$
710 IF E$="N"OR E$="n"THEN 800
720 INPUT "Which Data Pair to Edit? ";I
730 IF I<1 OR I>K THEN 700
740 GOTO 145
800 INPUT "Delete a Point <Y/N>? ";A$
805 IF A$="N"OR A$="n"THEN 900
810 INPUT "Which Point? ";N
815 K=K-1
820 FOR I=N TO K
825 X(I)=X(I+1):Y(I)=Y(I+1):Z(I)=Z(I+1)
830 NEXT I
835 INPUT "Delete Another Point <Y/N>? ";A$
840 IF A$="Y"OR A$="y"THEN 810
900 INPUT "Print New Input Table <Y/N>? ";A$
905 IF A$="N"OR A$="n"THEN 980
910 FOR I=1 TO K
915 P$=STR$(I)&" = "
920 PRINT #PN:PAUSE ALL
925 A$="X"&P$:PRINT #PN,A$;X(I)
930 A$="Y"&P$:PRINT #PN,A$;Y(I)
935 A$="Z"&P$:PRINT #PN,A$;Z(I)
940 PAUSE 0
945 NEXT I
950 PRINT #PN
980 INPUT "New Solution <Y/N>? ";A$
990 IF A$="Y"OR A$="y"THEN 200
999 STOP
1000 REM LINEAR SOLUTION FUNCTIONS
1010 F(0)=Z(L)
1020 F(1)=1
1030 F(2)=X(L)
1040 F(3)=Y(L)
1050 RETURN
```

ELLIPSE THROUGH FIVE POINTS - C. Rabe and P. Hanson. Bill Wilburn proposed this problem which comes from a quality control application. Consider the general equation of second degree which can define an ellipse, a parabola, a hyperbola or degenerate forms thereof depending on the magnitudes and signs of the coefficients:

$$(1) \qquad Ax^2 + Bxy + Cy^2 + Dx + Ey + F = 0$$

which has six arbitrary constants. Only five are independent since we can divide through by any one of them. You might think that we can arbitrarily select which one; for example, we could divide through by F. That would turn out to be a poor selection; it turns out that the result would be to preclude a solution based on five points which included the origin as one of the input points. In the TI-74 program which was developed we divided through by the coefficient of the x-squared term (A) to yield a modified general equation of second degree

$$(2) \qquad x^2 + B'xy + C'y^2 + D'x + E'y + F' = 0$$

where the primed coefficients are simply the unprimed coefficients from equation (1) divided by A. Consulting analytic geometry texts will reveal that

* If B is not equal to zero then the principal axis is not along a coordinate axis.

* B can be eliminated by a coordinate rotation where the rotation angle is $\tan 2\theta = B/(A-C)$.

* If B is equal to zero then the curve is an ellipse if A and C are of the same sign.

* The quantity $(B^2 - 4AC)$ is a constant independent of coordinate rotation.

* If $(B^2 - 4AC) < 0$ then the equation defines an ellipse.

* Those conditions can be applied to equation (2) by recognizing that $A' = 1$.

Analytic geometry texts also typically provide an equation for an ellipse with the major axis parallel to a coordinate axis of the form

$$(3) \qquad \frac{(x-h)^2}{a^2} + \frac{(y-k)^2}{b^2} = 1$$

where h and k are the coordinates of the center of the ellipse and a and b are the semiaxes. Again, there are four constants; thus, four points would be required to define such an ellipse. If we multiply through by a , expand the two quadratics, and collect coefficients we will have converted equation (3) to a form similar to the modified general equation (2):

$$(4) \qquad (1)x + (0)xy + (a /b)y + (-2h)x + (-2a k/b)y + (h -a + a k /b) = 0$$

Then, if the appropriate rotation has been performed on equation (2) such that the coefficient of the xy term is zero, then the remaining coefficients of equation (2) can be set equal to the corresponding coefficients of equation (4) and it can be shown that

$$(5) \qquad h = -D'/2$$

$$(6) \qquad k = -E'/(2C')$$

$$(7) \qquad a^2 = (D')^2/4 + (E')^2/(4C') - F'$$

$$(8) \qquad b^2 = a^2/C'$$

<u>Ellipse through five points</u> - (cont)

Based on the observations above the methodology selected to find the parameters of an ellipse through five points is to

1. Generate a set of linear equations (five equations in five unknowns, B' through F') by substituting the values of each x,y pair into equation (2). The rows of the matrix will be of the form

   $x_i y_i \quad y_i^2 \quad x_i \quad y_i \quad 1$, and the elements of the vector will be $-x_i^2$.

2. Solve for the coefficients of the general equation of second degree.

3. Rotate the coordinates to eliminate the xy term (set B = 0).

4. Convert the coefficients of the reduced equation of second degree to the parameters which define the ellipse.

5. Obtain the coordinates of the center of the ellipse in the original coordinate system by a coordinate rotation.

6. Thus, the five coefficients from the linear equation solution are transformed into the five constants a, b, h, k, and O. The program actually outputs 2a and 2b instead of a and b to meet a user preference.

A BASIC solution was developed for the TI-74 following those guidelines. Carl Rabe also developed a program in TRUE BASIC for a PC which includes a plot of the results. A solution on the TI-95 can be obtained by using the Linear Systems and Conic Sections programs of the Math module. A sample solution from the TI-74 for a problem provided by Bill Wilburn is at the right. The parameters Bill used to generate the five x,y pairs were h = 1, k = 2, 2a = 5.69, 2b = 3.56, O = 15 degrees.

```
X1 =  .8164
Y1 =  3.7844

X2 =  1.3849
Y2 =  3.861

X3 =  1.9751
Y3 =  3.8571

X4 =  2.5946
Y4 =  3.7433

X5 =  3.2724
Y5 =  3.4122

A =  1
B = -.7059885942
C =  2.224120745
D = -.5841439941
E = -8.199769656
F =  1.1696526

Angle =  14.98669461

maj =  5.690143483

min =  3.555934006

h =  .9987252202

k =  2.001882832

hr =  1.482430831
kr =  1.67552552
```

A listing of a BASIC program for the TI-74 is on page 19. It turns out that the solution will yield divide-by-zero errors for some cases of zero inputs for either X or Y. The problem is related to an inadequacy in the linear equation solution, in particular the divides at lines 335 and 340. That inadequacy is discussed in more detail on page 2 of this issue. It turns out that for the ellipse through five points problem the difficulty can be circumvented by selecting a sequence of data entry such that the diagonal elements of the matrix can not be zero. Lines 101 through 103 of the program define the rules for the data entry sequence. Lines 155-160 and 175-180 reject data entry which is not consistent with the rules. The program includes two other special features:

* There is no DIM statement even though subscripts are used. Thus, the program relies on the builtin BASIC capability to handle subscripts up to ten.

* The definitions of the angle at line 515 as ATN(1.E+100)/2 and at line 595 as ATN(1.E+100) makes the calculations independent of whether the TI-74 is in the degrees or radians mode. If the user wants to limit the solution to the degree mode he can change those angles to 45 degrees and 90 degrees.

Ellipse through five points - (cont)

```
10 REM Ellipse through 5
 Points
20 REM 7 July 1990
100 REM Data Entry
101 PRINT "Program will
not accept zeroes":PAUSE
 2
102 PRINT "for X1, Y1, Y
2, X3 or Y4.":PAUSE 2
103 PRINT "Adjust order
of data as needed.":PAUS
E 2
105 INPUT "Use Printer <
Y/N>? ";A$
110 IF A$="Y"OR A$="y"TH
EN PN=1 ELSE 140
115 INPUT "Device Code ?
 ";P$
120 OPEN #1,P$,OUTPUT
135 E$=""
140 FOR I=1 TO 5
145 P$=STR$(I)&" = "
150 A$="X"&P$:INPUT A$;X
(I)
155 IF I=1 OR I=3 THEN 1
60 ELSE 165
160 IF X(I)=0 THEN PRINT
 "X"&STR$(I)&" cannot be
 0":PAUSE:GOTO 150
165 IF PN<>0 THEN PRINT
#PN,A$,X(I)
170 A$="Y"&P$:INPUT A$;Y
(I)
175 IF I=3 OR I=5 THEN 1
85 ELSE 180
180 IF Y(I)=0 THEN PRINT
 "Y"&STR$(I)&" cannot be
 0":PAUSE:GOTO 170
185 IF PN<>0 THEN PRINT
#PN,A$,Y(I)
190 PRINT #PN:IF E$<>""T
HEN 850
195 NEXT I
200 REM Set Up Linear Eq
uations
210 PRINT "Solving"
220 FOR I=1 TO 5
230 A(I,1)=X(I)*Y(I)
240 A(I,2)=Y(I)*Y(I)
250 A(I,3)=X(I)
260 A(I,4)=Y(I)
270 A(I,5)=1
280 B(I)=-X(I)*X(I)
290 NEXT I
300 REM Linear Equation
Solution
310 N=5
320 FOR L=1 TO N
325 P=A(L,L)
330 FOR J=L TO N
335 A(L,J)=A(L,J)/P:NEXT
 J
340 B(L)=B(L)/P
345 FOR I=1 TO N
350 IF I=L THEN 375
355 G=A(I,L)
360 FOR J=L TO N
365 A(I,J)=A(I,J)-G*A(L,
J):NEXT J
370 B(I)=B(I)-G*B(L)
375 NEXT I
380 NEXT L
385 B(0)=1
390 D=B(1)*B(1)-4*B(2)
400 REM Print Linear Equ
ation Solution
410 G=65
420 FOR I=0 TO 5
430 X$=CHR$(G+I)&" = "
440 PRINT #PN,X$;B(I)
450 IF PN=0 THEN PAUSE
460 NEXT I
470 PRINT #PN
500 REM Coordinate Rotat
ion
503 IF D<0 THEN 510
505 PRINT #PN,"Not an El
lipse":IF PN=0 THEN PAUS
E
507 PRINT #PN:GOTO 800
510 IF B(1)=0 THEN T=0:G
OTO 525
515 IF B(0)=B(2)THEN T=A
TN(1.E+100)/2:GOTO 525
520 T=(ATN(B(1)/(B(0)-B(
2))))/2
525 CSQ=COS(T)*COS(T)
530 SSQ=SIN(T)*SIN(T)
535 SXC=SIN(T)*COS(T)
540 AP=B(0)*CSQ+B(1)*SXC
+B(2)*SSQ
545 C=(B(0)*SSQ-B(1)*SXC
+B(2)*CSQ)/AP
550 D=(B(3)*COS(T)+B(4)*
SIN(T))/AP
555 E=(-B(3)*SIN(T)+B(4)
*COS(T))/AP
560 F=B(5)/AP
565 HR=-D/2
570 KR=-E/(2*C)
575 AA=D*D/4+E*E/(4*C)-F
580 MAJ=AA:MIN=AA/C
585 H=HR*COS(T)-KR*SIN(T
)
590 K=HR*SIN(T)+KR*COS(T
)
595 IF MIN>MAJ THEN MAJ=
MIN:MIN=AA:T=T-SGN(T)*AT
N(1.E+100)
600 REM Output
605 PRINT #PN,"Angle = "
;T
610 IF PN=0 THEN PAUSE
615 PRINT #PN
620 PRINT #PN,"maj = ";2
*SQR(MAJ)
625 IF PN=0 THEN PAUSE
630 PRINT #PN
635 PRINT #PN,"min = ";2
*SQR(MIN)
640 IF PN=0 THEN PAUSE
645 PRINT #PN
650 PRINT #PN,"h = ";H
655 IF PN=0 THEN PAUSE
660 PRINT #PN
665 PRINT #PN,"k = ";K
670 IF PN=0 THEN PAUSE
675 PRINT #PN
700 REM Print Rotated Ce
nter
710 PRINT #PN,"hr = ";HR
720 IF PN=0 THEN PAUSE
730 PRINT #PN,"kr = ";KR
740 IF PN=0 THEN PAUSE
750 PRINT #PN
800 INPUT "Edit Input Da
ta <Y/N>? ";E$
810 IF E$="Y"OR E$="y"TH
EN 820 ELSE 900
820 INPUT "Which Data Pa
ir to Edit? ";I
830 IF I<1 OR I>5 THEN 8
20
840 GOTO 145
850 INPUT "Edit Another
Point? ";A$
860 IF A$="Y"OR A$="y"TH
EN 820
900 INPUT "Print Input D
ata <Y/N>? ";A$
905 IF A$="N"OR A$="n"TH
EN 955
910 FOR I=1 TO 5
915 P$=STR$(I)&" = "
920 PRINT #PN:PAUSE ALL
925 A$="X"&P$:PRINT #PN,
A$;X(I)
930 A$="Y"&P$:PRINT #PN,
A$;Y(I)
940 PAUSE 0
945 NEXT I
950 PRINT #PN
955 INPUT "Solve with Ed
ited Data <Y/N>? ";A$
960 IF A$="Y"OR A$="y"TH
EN 200
980 INPUT "New Problem <
Y/N>? ";A$
990 IF A$="Y"OR A$="y"TH
EN 135
999 END
```

ADDENDUMS FOR THE TI-74 MATH MODULE - Ron Burnham writes that he found three addendums with his TI-74 Mathematics Library Guidebook. I only received one which had the notation 1059977-0101 in the lower right hand corner. Copies of the two which I did not have are reproduced below.

Addendum: TI-74 Math Library Cartridge (Continued)

Gamma Function

See pg. 2-8

The Gamma Function Program (GAM) can calculate answers for negative non-integers. The magnitude of the answer is correct, but the sign of the answer is not always correct. To avoid calculating an incorrect answer for gamma, use the following program instead of the program built into the cartridge. This program calls a subprogram that is in the Math Library Cartridge.

```
200 RAD: INPUT "Enter X: ";X
210 CALL GAMS(X,G)
220 IF G>294.73 THEN 230 ELSE 240
230 DISPLAY "Ln(Gamma(X)) = ";G:PAUSE:GOTO 200
240 GM = SGN(X)^INT(X)*EXP(G)
250 DISPLAY "Gamma(X) = ";GM:PAUSE:GOTO 200
```

To calculate gamma:

1. Enter the program.
2. With the Math Library cartridge installed, run the program.
3. Enter X as prompted.
4. View the result.
5. If you have more gamma calculations to perform, press [ENTER] and return to step 3.
6. If you are finished with gamma calculations, press [BREAK] to stop the program.

Runge Kutta

The Runge-Kutta Program requires that you enter a step size that causes no long or short step to occur at the end of the interval. Otherwise, the program calculates the last step incorrectly. To determine a step size that causes all steps to be equal in width, choose an integer number of steps to be between Xmin and Xmax and use that number in the following equation.

Step size = (Xmax − Xmin)/number of steps

Addendum 2: TI-74 Math Library Cartridge

Complex System Program

The COMPS program may return the message MATRIX IS SINGULAR even though there is a solution for a specific matrix.

If this occurs, you can solve your complex system using option 5 of the Matrices (MAT) program. Using the method shown below, manually expand your n × n system of complex equations into a 2n × 2n system of real equations.

Each complex coefficient is evaluated as

$$(a_{11}, b_{11}) = \begin{matrix} a_{11} & -b_{11} \\ b_{11} & a_{11} \end{matrix}$$

Using this method, the system

$$\begin{matrix} (a_{11}, b_{11}) & (a_{12}, b_{12}) & (a_{13}, b_{13}) & & (c_1, d_1) \\ (a_{21}, b_{21}) & (a_{22}, b_{22}) & (a_{23}, b_{23}) & & (c_2, d_2) \\ (a_{31}, b_{31}) & (a_{32}, b_{32}) & (a_{33}, b_{33}) & & (c_3, d_3) \end{matrix}$$

is expanded to

$$\begin{matrix} a_{11} & -b_{11} & a_{12} & -b_{12} & a_{13} & -b_{13} \\ b_{11} & a_{11} & b_{12} & a_{12} & b_{13} & a_{13} \\ a_{21} & -b_{21} & a_{22} & -b_{22} & a_{23} & -b_{23} \\ b_{21} & a_{21} & b_{22} & a_{22} & b_{23} & a_{23} \\ a_{31} & -b_{31} & a_{32} & -b_{32} & a_{33} & -b_{33} \\ b_{31} & a_{31} & b_{32} & a_{32} & b_{33} & a_{33} \end{matrix} \times \begin{matrix} X_1 \\ Y_1 \\ X_2 \\ Y_2 \\ X_3 \\ Y_3 \end{matrix} = \begin{matrix} c_1 \\ d_1 \\ c_2 \\ d_2 \\ c_3 \\ d_3 \end{matrix}$$

Load the values from the manual expansion of your system into the Matrices (MAT) program, option 5, to solve the matrix. The results of the MAT program should be interpreted as follows:

- Odd-numbered results (X_1, X_3, X_5, etc.) are the real parts (X_1, X_2, X_3, etc.).
- Even-numbered results (X_2, X_4, X_6, etc.) are the corresponding imaginary parts (Y_1, Y_2, Y_3, etc.).

TI PPC NOTES

NEWSLETTER OF THE TI PERSONAL PROGRAMMABLE CALCULATOR CLUB

P. O. Box 1421, Largo, FL 34649

Volume 14, Number 5 Final Issue November 1991

For everything there is a season,
and a time for every matter under heaven,
a time to be born, and a time to die, ...

That passage from Ecclesiastes 3 is pertinent to this last issue in two ways. First, I am sorry to report that Carl Rabe died of a cerebral hemorrhage due to a heart attack. I never met Carl face-to-face, but considered him as a good friend. I talked to him several times on the telephone and accumulated a lenghthy file of letters on programs and other subjects of common interest. Members will recall that Carl was instrumental in the fine tuning of the smallest circle analysis.

Second, it is time to publish the last issue of our newsletter. I am even later than usual with this final issue, primarily due to difficulty in generating much enthusiasm for a final issue. Several members have asked about the possibility of continuing some level of support in the coming year. My present plans are to:

1. Maintain the post office box address through mid-1992. During that time I will continue to provide magnetic tape service and magnetic card service, will provide back issues if ordered, and will answer correspondence as time permits.
2. Make up compilations of material from the V14 issues for the TI-74, the TI-95 and the TI-81 and offer them to past purchasers.

My personal plans are not well defined. I still like to work on problems whether of the programming variety or of the more basic mathematical kind, so if you have a vexing problem and would like some assistance feel free to write. If the problem captures my imagination I will try to help. I also hope to write some technical papers on some of my work in the early days of inertial navigation.

Finally, I want to thank all of those who have supported the club through the years through continuing membership, submission of programs, and the like. May you all have a Happy Holiday Season.

ERRATA

V14N4P4-7 Polynomial Curve Fit on the TI-81 - George Thomson notes that once again I failed to point out that the solution was done by the method of least squares minimizing errors in Y, and with the X values assumed to be error free and of equal weight. George also wondered how many readers really know how tricky polynomials can be because of the "hill-and-valley" problem. The equation may fit well at or near the observed data points but have ridiculous values between the points. One way to observe problems of this sort is to overlay the plot of the function on the input data as was done in the TI-81 program.

One final personal note: if you look closely at nature you will find that the Lord tended to use logarithmic and exponential functions and statistical methods - He didn't have much use for polynomials.

HARDWARE AND SOFTWARE WANTED - HP-35, HP-45 and HP-65 together with the original documentation (owner's manual, programming guide, handbook, etc.) and external power adapter. Slide rules of any configuration plus books and pamphlets on such items. Also interested in programs which calculate residential heat loads; e.g., to solve for in-house temperatures given various ambient outside temperatures, solar inputs, interior thermal mass and other related parameters. This would be used to size construction parameters for a solar heated greenhouse or solar heated living spaces. Desireable, but not mandatory, that programs will run on an HP-45, HP-65 or TI-59. Write to William S. West, 731 Monroe Street, Apartment 303, Rockville MD 20850-2716 .

TI-66/PC-200 SALE - Page 19 of the August 1991 DAMARK catalog offers a TI-66 and PC-200 for a combined price of $39.99 plus $6.00 for shipping and handling. Ask for Item No. B-474-182785. Write to 7101 Winnetka Ave. N., P. O. Box 29900, Minneapolis MN 55429-0900 or call 1-800-729-9000.

CONTINUING SUPPORT FOR THE CC-40 - The Lima User Group emphasizes support of the TI-99/4, but is also a source of continuing information on the CC-40 including articles on hardware availability and software. Examples of the kind of material available appear in the article on the Wafertape in V14N4P12-13 and the data bank program on page 3 of this issue. Membership dues are $15.00 per year in the United States, $20.00 per year overseas. Members receive ten issues of the newsletter *Bits Bytes & Pixels* each year. Send your subscription to the Lima User Group, Box 647, Venedocia OH 45894.

All back issues that contain CC-40 articles are available for three cents per page. Members are billed for the cost of back issues they request after the material is sent. The CC-40 software library consists of 24 CC-40 BASIC programs, five of which were typed in from *TI PPC NOTES*. The entire collection is available to any member who sends a QuickDisk or Wafertape and a paid return mailer.

Another example of the kind of support that is available is a video tape in which the CC-40 PLUS, with built-in cassette interface is shown. The CC-40 PLUS is a never released product that was under development by TI. The tape also shows the never released TI-99/2 which was designed to compete with the Timex/Sinclair 1000. The price for the tape is five dollars. It runs for six hours, but only an hour or so is devoted to the CC-40 and TI-99/2. The tape was made at the Lima User Group's May 1991 TI-99/4A conference.

__USING THE CC-40 (and the TI-74) AS A PORTABLE DATA BANK__ - Charles Good. Have you seen those electronic "organizers" in the department stores? They are battery operated dedicated computers that look like large calculators. Marketed under names such as "THE BOSS" or "THE ORGANIZER", they are designed to store lists of names and addresses or hour by hour appointment calendars. With these devices it is easy anywhere and anytime to sort through a large data base for a particular name, address, or phone number. With many of these devices you can download their data to a desktop PC or print the information on a printer.

Well, you can do the same thing with a CC-40. Since the CC-40 has CMOS RAM, any BASIC program entered into the CC-40 stays in memory even after the computer is turned "off". BASIC programs will remain in the CC-40's memory for many months in the "computer is turned off" mode before the batteries finally need to be changed. Short appointment calendars, notes or address lists can be stored in the CC-40 as text that is displayed when any of the ten user programmable hot keys are pressed from command mode. Data can also be stored as part of a memory resident BASIC program, but there is a potential problem with this method. Although the CC-40 retains a BASIC program in memory when it is turned "off", it does NOT retain any of the numeric or string variables generated by the BASIC program. There are CC-40 programs in my software library that allow you to open a disk (or wafertape) file and read in a list of names, addresses, and phone numbers. When you do this and then later turn the CC-40 "off" to conserve battery power, the BASIC program remains in memory. However, the data read in from disk and stored in strings is GONE! There is certainly no advantage to the CC-40's portability if you have to go to a disk or tape drive every time you want to look up someone's phone number! The solution to this problem is to store your information as an actual part of the BASIC program in line numbers with DATA statements. When stored in this way, your data is retained in the CC-40's memory even when the computer is "off".

I wrote a CC-40 program to deal with this situation. A somewhat similar CC-40 program was published in the Volume 1 Number 2 issue of *ENTHUSIAST 99*, but it doesn't work on many CC-40's because it POKES to an absolute memory address. My program works! It stores a name/address/phone number list as data statements. Only one person's data at a time is read into string variables; thus, the CC-40's memory is not wasted storing the entire data bank twice, once in data statements and a second time in strings. From the running program you can view all the data or search for the data of one particular person by inputting the person's last name. Because of the POS statement in line 200, you don't even have to enter the entire last name for a search. All you need to enter is a text string that is contained within the last name, such as inputting BUCK when searching for the last name ARBUCKLE. If the data base includes more than one person with the same last name the program will still help you find the information you want. An option also permits the user to view all input in sequence. The data for each person is displayed on a single 80 column line. You can scan left and right across this line of data at your leisure using the CC-40's arrow keys.

You can store about 100 program lines of names, addresses, and "other information" as DATA statements with this program in a minimum 6K CC-40. It takes just a few seconds to find the last of the 100 DATA statements in a name search. Editing is easy. From the CC-40's command mode just bring up the appropriate line number containing the DATA statement and type over or add to the existing DATA of that line number. To add more names to the data bank, just create more BASIC line numbers for the additional data. To obtain a hard copy of the data you can LIST the program to a HexBus compatible printer. You can also use the HexBus RS232 peripheral to list the program with all of your DATA to a non HexBus printer or dump the program (via a cable linking the HexBus RS232 to the TI-99/4A RS232) directly into a TI-99/4A.

Although you can't put a CC-40 in your pocket as can be done with many modern electronic organizers, you can easily put the "smaller than most books" CC-40 in a small briefcase or purse. And you do so much more with the CC-40! Unlike most of the modern "organizers" the vintage 1983 CC-40 is a portable and truly flexible programmable computer. "Modern" does not always mean "better". (Editor's Note: The listed program will operate equally well in either a TI-74 or CC-40. Members will recall that equivalent limitation on storage of data in the TI-74 was discussed in V12N4P20.)

```
100 REM TI-74 DATA BANK FILE
110 REM by Charles Good, Lima, Ohio User Group, June 1991
120 PRINT "  --Name/Phone/Address File--":READALL=0:PAUSE 1
130 INPUT "Read All Names? (Y/N) ";YN$
140 IF YN$="Y"THEN READALL=1:GOTO 180
150 IF YN$="y"THEN READALL=1:GOTO 180
160 PRINT "USE UPPER CASE TO ":PAUSE .5
170 INPUT "ENTER DESIRED LAST NAME- ";INPUT$
180 READ FN$:IF FN$="END"THEN PRINT "END OF FILE":PAUSE 1:RESTORE 1000:GOTO 120
190 READ LN$,REST$:IF READALL=1 THEN 250
200 IF POS(LN$,INPUT$,1)=0 THEN 180
210 INPUT "Is the person "&FN$&" "&LN$&"? ";YN$
220 IF YN$="Y"THEN 250
230 IF YN$="y"THEN 250
240 GOTO 180
250 PRINT FN$&" "&LN$&" "&REST$:PAUSE
260 IF READALL=1 THEN 180
270 RESTORE 1000:GOTO 120
970 REM FIRST NAME, LAST NAME, OTHER INFORMATION such as phone number and address
980 REM Use ONLY UPPER CASE for first and last names. Commas are required after
990 REM the first and last name. Use no commas in the other information field.
1000 DATA BARBARA,GOOD,616-857-2256 11 Lakeshore Dr. Douglas MI 49406
1010 DATA IAN,GOOD,419-667-3131 15276 Main Venedocia OH 45894
1020 DATA JACK,TURNER,Chestnut Lane Douglas MI 49406
5000 DATA END
```

AN ANOMALY IN THE LINEAR EQUATIONS SOLUTION ON THE TI-81, TI-68 AND TI-74

Timothy Baumgartner writes: "My Pre-calculus students have been using the TI-81 to solve matrix equations representing systems of linear equations. To introduce the topic of inconsistent systems I asked them to solve the system

$$3x + 4y = 7$$
$$6x + 8y = -2$$

hoping that they would get an error message after entering the matrices

$$[A] = \begin{vmatrix} 3 & 4 \\ 6 & 8 \end{vmatrix} \quad \text{and } [B] = \begin{vmatrix} 7 \\ -2 \end{vmatrix}$$

and entering the expression [A] [B] . Instead, they saw the display

```
[ -1.066666667E13]
[ 8E12           ]
```

Upon investigating, we found that the determinant of [A] had been calculated as -6E-12 instead of the correct value of zero. This should be a very straightforward calculation:

$$\det[A] = 3 \times 8 - 6 \times 4 = 0$$

My question is this: How does the TI-81 calculate its determinants?"

Editor's Note: I do not know the exact sequence of calculations used to calculate determinants and solve linear equations in the TI-81. The method used is certainly not the simple one you describe since it will not work for systems of higher order. Of course, one can use the old artifice for a 3rd order system of rewriting the first and second columns to the right of the third column, and then doing the diagonal calculations, but as my old college algebra text says "... this method of expansion applies to a determinant of third order but not to one of higher order." Also, almost certainly the TI-81 does not evaluate a determinant by brute force expansion by minors. More probably it accomplishes the task through appropriate matrix manipulations such as additions and subtractions of one row (or column) from another such that zeroes are generated for selected elements of the matrix. Those methods typically involve multiplying or dividing one row (or column) by one of the elements of another row (or column). For example, suppose we want to place a zero in the first element in the second row. One way to do it would be to

(1) Divide each element of the first row by the first element of the first row.

(2) Multiply each element of the result from (1) by the first element of the second row.

(3) Subtract the result from (1) and (2) element by element from the elements of the second row.

Errors can be introduced if divisions occur which result in answers with repeating decimals. The calculations for the replacement values for the second row for your particular problem follow:

(1) Start with the elements of the first row, [3,4] and divide each element by 3, yielding [1,4/3] where the important point is that the second element is not exactly 4/3 as shown, but is actually 1.3333333333333 (thirteen decimal digits).

(2) Multiply [1,1.3333333333333] by 6 yielding [6, 7.9999999999998].

An Anomaly in the Linear Equations Solution on the TI-81, TI-68 aned TI-74 - (cont)

(3) Subtract [6,7.9999999999998] from the second row of the original matrix yielding the matrix

$$\begin{vmatrix} 3 & 4 \\ 0 & 2E-12 \end{vmatrix}$$

(4) Evaluate the matrix as 3 x 2E-12 - 0 x 4 = 6E-12. That answer is the same magnitude as the answer from the TI-81, but of different sign, suggesting that our algorithm is similar, but not identical to that in the TI-81.

What if we interchange the variables x and y in the original problem statement? Then the columns of the matrix would be interchanged. For that problem the TI-81 calculates the determinant as exactly zero, and an attempt to solve the linear equations results in the error message "ERROR 05 MATH".

Tests of other linear equation solvers show that for the test problem:

* The ML-02 program in the Master Library module for the TI-59 finds the determinant to be exactly zero. If the user then tries to solve the linear equations, which violates the instructions on page 10 of the manual, then the solution is returned as -1.6666667E99 and 9.9999999E99 where the second output is flashing to indicate an error has occurred. If the columns are reversed the results are the same.

* The linear equations solution of the TI-68 does not display the determinant. Page 7-6 of the manual states "If the system is singular, an error condition occurs." An attempt to solve the test problem yields the solution X1=1.0666667E13, X2= -8E12, the same magnitudes as with the TI-81, but with opposite signs. If the columns are reversed the message "Error" appears in the display.

* The linear equations solution portion of the Mathematics module for the TI-74 finds the determinant to be -6.E-12 and the solutions of the linear equations to be X1 = -1.066667E+13 and X2 = 8.E+12, the same results as those obtained with the TI-81. If the columns are reversed the message "THE SYSTEM IS SINGULAR" is returned.

* The linear equations solution portion of the Mathematics module for the TI-95 finds the determinant to be exactly zero. An attempt to solve the linear equations yields the message "SINGULAR". If the columns are reversed the determinant is still zero and an attempt to solve the linear equations results in the "SINGULAR" message.

There is no discussion of the method of solution in the guidebooks for the TI-68, TI-81 or TI-95 Mathematics Library. Pages 5-10 and 5-11 of the *TI-74 Mathematics Library Guidebook* states that the lower-upper decomposition method is used to find the determinant and to solve the linear equations. The detailed description is nearly identical with that which appeared on page 13 of the manual for the Master Library module for the TI-59. The reference in both the TI-59 and TI-74 manuals is *Numerical Methods*, Germund Dahlquist and Ake Bjork, Prentice Hall, 1974.

ANOTHER SOURCE OF CC-40 HARDWARE - Charles Good writes: "A source of used CC-40 computers, peripherals, and user manuals (including the Wafertape manual) is Jim Lesher, 722 Huntley, Dallas TX 75214. Telephone 214-821-9274. I purchased a used 6K CC-40 with all of the original documentation for $50.00 plus shipping. Prices are subject to change at any time."

EXPERIENCE WITH THE PC INTERFACE CABLE - Greg Lind writes: I recently purchased the PC-Interface for the TI-74 and TI-95. Since I own a TI-74 I'll be referring to that device. This interface lets you connect your TI-74 to any IBM or compatible computer. With the interface connected you now have disk drives for storage, an 80 column printer, and the computer screen for displaying output or program lines. This is now all connected to the TI-74 as if it was all one unit. It is really easy to use. Here's how it works. The interface connects to your parallel printer port. TI uses this port for rapid transmission of data and it's very fast. The only drawback is if your PC has only a single parallel port and your printer needs the parallel port as well. You have to disconnect the interface if you want to use the printer. I made a cable extension to bring the port connection to the front of the computer instead of always trying to reach behind my computer to connect the interface or printer cable.

Once the interface is connected you type PCIF at the DOS prompt. Now your TI-74 has your IBM or compatible at its disposal. Each access device has a special number:

Computer parallel printer -------------	14
Computer screen -----------------------	45
Computer disk for program storage -----	100
Computer disk for data storage or ASCII format for text editor ----------	101

Device "14" is for the parallel port if you have two ports. If not, all you do is save in an ASCII format and use a word processor program or anything similar to print to the printer. For example LIST "14.R=C" where R=C is the printer option would dump the entire program to the printer. You can limit the listing to certain line numbers with a command such as LIST "14.R=C",150-200 which will then print only lines 150 through 200. You can print output from a program as well.

Device "45" is for the computer monitor screen. You can use this to list program lines; for example LIST "45.P=Y" where P=Y is also an option. This stops the output when the screen is full until you press ENTER. You can also use device "45" in a program to move text from your program to the monitor. For example:

```
10 A$="TEXAS INSTRUMENTS, PC INTERFACE"
20 OPEN #2,"45.P=Y",VARIABLE 80, OUTPUT
30 PRINT A$
40 CLOSE #2
```

By changing the information inside the quotes in line 20 to "14.R=C" you would send the string to the printer if you had two ports on your PC.

Device "100" is used to store your programs to disk and also to retrieve them. SAVE "100.filename.xxx" is the basic option to save your programs where the filename.xxx portion of the statement would equal the name you give to your program with an extension. For example: SAVE "100.program.doc" stores a program. To retrieve the program you would enter OLD "100.program.doc" on your TI-74.

Device "101" is used for data storage and to save your program in ASCII format. When saved in ASCII format you can use a text editor to make changes or to print the program. You can also save data that the program produced while running. For example:

```
10 A$="TEXAS INSTRUMENTS, PC INTERFACE"
20 OPEN #2,"101.filename.xxx",VARIABLE 80,OUTPUT
30 PRINT #2,A$
40 CLOSE #2
```

will save A$ to the disk for storage. To retrieve it later use the program

Experience with the PC Interface Cable - (cont)

```
20 OPEN #2,"101.filename.xxx",VARIABLE 80,INPUT
30 INPUT #2,A$
35 PRINT A$
40 CLOSE #2
```

LIST "101.filename.xxx" would save it in the ASCII format for a word processor to read.

The nice part is that you can take any program written in, let's say GW-BASIC and run it on the TI-74. You have to change some things. For example, TI-BASIC doesn't use WHILE and WEND statements. These would have to be changed to IF-THEN-ELSE statements. First you need any of the programs in the ASCII format. You need to save the TI-74 programs by using LIST "101.filename.xxx". You need the TI program to convert it to a binary format for the TI-74 to read. When you save your text from the text editor you save it with B74 as the extension, i.e., as filename.B74. Then at the DOS prompt you type TIC74 filename.B74 and the program does its thing and stores the new binary form as "filename.PGM". To retrieve this you type OLD "100.filename.PGM" on the TI-74 and presto you have a working program and you did not have to use the TI-74 keyboard to enter it. This also works for programs written in Pascal. You can also direct output to the computer screen for all of the software cartridges, i.e., Chemical Engineering, Finance, Mathematics, Statistics and Learn Pascal. In conclusion, the interface is a worth while piece of equipment for use with your TI-74 if you have an IBM compatible computer.

A PUZZLE - Dan Mahony. On the parallel sides of a river which is five miles wide are located the villages A and B. Village B is located 4 miles north of a point directly opposite village A. For a width of 3 miles on village A's side of the river the current flows south at 2 mph. The rest of the channel flows north at 5 mph. You can row your boat at 7 mph. You wish to row from A to B using a constant heading relative to north. What is that angle? What is the time of transit?

The heading is 56.4602... degrees.
The time of transit is 0.8569691599... hours,
or exactly $(25\sqrt{1993} - 80)/1209$ hours.

George Petersen notes that the time of transit should be able to be reduced if heading changes are allowed during the transit. He asks "What is the minimum time of transit?" Send your solution to the editor who will forward it to George.

MORE ON PC-324 BATTERY USEAGE - V14N4P11 reported that the editor's PC-324 seemed to be a "battery eater". Members Greg Lind, Scott Garver and Bill Wilburn reported similar experience. They noted that since the PC-324 has no ON/OFF switch it is always in the ON state and there must be some power drain. I also found the following note at the bottom of page 4 of the *PC-324 User's Guide:*

> "We recommend that you use alkaline batteries in your printer. If non-alkaline batteries must be used, remove the batteries when you are not using the printer."

I wrote to TI for an interpretation. Tom Ferrio called to confirm a small drain when the printer is not is use. TI expects a set of alkaline batteries should last 6 to 12 months.

Graphing on the TI-81 and fx-7000G - (cont)

As an exercise, you should convince yourself that the specific characteristics of the graph are consistent with the function values as read from the Y variable. You should also convince yourself that the rounding which occurs in the x-squared values is consistent with the resulting function values which are in the Y variable. To assist you with that I have included a table which presents the exact x-squared values for the seven x values which appear in the table on page 13:

x	x^2
2.000000684211	4.000002736844 468144692521
2.000000789474	4.000003157896 623269196676
2.000000894737	4.000003578948 800554299169
2.000001	4.000004000001 000001
2.000001105263	4.000004421053 221606299169
2.000001210526	4.000004842105 465373196676
2.000001315789	4.000005263157 731300692521

"Hamming's Law of logarithms" - Charlie Williamson called my attention to the interesting observation that the logarithm to the base 2 of a number is equal to the sum of the natural logarithm of the number and base 10 logarithm of the number, within one per cent. Charlie noted that Volume 1 of Knuth's *The Art of Computer Programming* ascribes the observation to Hamming, but that he had personal correspondence suggesting that Hamming may not have been the originator. I looked in my copy of Knuth's book. I found no reference to Hamming! After some additional correspondence we realized that, in the first edition, which Charlie uses, Exercise 22 on page 25 states:

22. [20] Prove "Hamming's law of logarithms":

$$\log_2 x \approx \ln x + \log_{10} x,$$

with less than 1% error! (Thus a table of natural logarithms and of common logarithms can be used to get approximate values of binary logarithms as well.)

But, in the second edition, which the editor uses, the problem begins as:

22. [20] Prove that

$$\lg x \approx \ln x + \log_{10} x, \quad \ldots$$

where we note that the reference to Hamming has been deleted and the notation lg x is introduced for the binary logarithm. Of course, the mystery is, if Hamming wasn't the source of the relationship, then who was?

REPAIR OF TI-74/TI-95 PERIPHERALS - P. Hanson. I had a still unexplained double failure of both my CI-7 Cassette Interface and PC-324 Printer. I called 1-800-TI-CARES for repair information. Repair service is not available for the CI-7 so I ordered a replacement from EduCALC for $26.95 plus $2.00 for shipping. Repair service is available for the PC-324. The cost is $45.50 plus state sales tax plus $4.00 for shipping and handling, substantially less than the $89.95 listed for a new unit in the EduCALC catalog. TI reserves the right to either repair the returned unit or to replace it with a reconditioned unit. I shipped my defective unit on October 2 and received a reconditioned unit on October 22.

Function Analysis on the TI-95 - (cont)

The following table lists part of the generated data base for the Gauss program to thirteen digits so that you obtain a precision partial check of your Gen program. For n=2, R026 = 0.5773502691895 , and R027 0.9999999999987 .

n	W = 0 n<512	+D1 in registers	W = 1 n=512	+D2 in registers
4	R028	0.8611363115941	R028	0.9999889909844
4	R029	0.3478548451378	R029	0.00002825263731350
4	R030	0.3399810435849	R030	0.9999419946068
4	R031	0.6521451548624	R031	0.00006576573169996
8	R032	0.9602898564976	R032	0.9998574463700
8	R033	0.1012285362912	R033	0.00011033319034104
256	R532	0.01837081847882	R532	0.02145312295978
256	R533	0.01224583436957	R533	0.006128523194201
256	R534	0.006123912375182	R534	0.01532423508490
256	R535	0.01224767164009	R535	0.006129214171627

Acknowledgements

The H. R. Beck book contains programs which are the basis for the Romberg and Gen routines. The Gen routine was speeded up using an initial estimate formula from *Methods of Numerical Integration* by Davis and Rabinowitz. The differentiation routine is from *The Sourcebook for Programmable Calculators*.

FAST FACTOR FINDING - Larry Leeds writes "The program UBASIC contains a large number of functions. For my purposes the functions for FACTORS and PRIMES are the important ones. Many of the others are Greek to me but just what a professor of mathematics would appreciate. One function for PI returns any selected number of digits up to 2500. Another solves the Tower of Hanoi (12 disks) which requires 4095 moves. On my machine this runs so fast that the display is almost a blur. It took the 386SX about one minute to factor

9999999999999999999999999999999999988 (that is, 34 nines followed by 2 eights) into

2 * 2 * 11 * 23 * 1774093100831 * 556984462668427315679

For another test I multiplied together three eleven digit primes to obtain

778438393067011463161307972470183. It took about 75 seconds to find the factors

99999999907 * 99999999977 * 77843839397

And we worked so hard to get a speedy factor finder for the TI-59!"

A FALSE ALARM? - V13N3P22 and V13N4P25 discussed the indications in the 8K RAM documentation that the battery life may limit the service life of the RAMs to three to five years. In a recent telephone conversation with Hewlett Ladd we noted that we have had many modules in operation since May 1987 and have seen no indication of failure to date, four and one-half years later.
