TI PPC NOTES

NEWSLETTER OF THE TI PERSONAL PROGRAMMABLE CALCULATOR CLUB

P. O. Box 1421, Largo, FL 34649

Volume 13, Number 1 First Quarter 1988/1989

Welcome back for another "year" in the life of our newsletter. This issue is later than usual for a number of reasons. First, I have been busy at work, and as I have explained before, making a living has to take priority. Second, just as I was getting ready to go to press I discovered that there was an error in one of the key programs, the solution to the circle problem proposed on the last page of the last issue. Since I hate to publish bad programs I had to define a revised solution, program it and check it out, and rewrite the accompanying text.

What will you find in this issue? Primarily you will find responses to requests which appeared in the comments section of the remewal form. Thus, you will find several articles on translation, both from the TI-59 to the TI-95, and from the TI-59 or TI-95 to BASIC for the TI-74. You will also find a compiled listing of all the programs from the old PPX Exchange which are available from other members. You will also find descriptions of two bugs which occur with the TI-95, with a work-around for one of them. Finally, we continue to list hardware for sale, sources of paper and supplies, and "want ads" for hardware.

If you want a programming challenge for the holidays, you might try to add a plot routine to the biorhythms program for the TI-74, similar to that which was available with LE-21 on the TI-59 and PC-100.

TABLE OF CONTENTS

Happy Holidays

ERRATA

Biorhythms in Programmable Calculator News - Becky Johnson of TI reports that there were two errors in the listing of the Biorhythms program on page 4 of the Volume 2, Number 2 issue of Programmable Calculator News. Lines 0210 and 0338 should read:

```
0210   + RCL C + 31 X RCL B

0338   PRT SBL E' STO F (
```

where the errors were a missing multiply sign in line 0210 and SLB instead of SBL in line 0338.

Errors in Extended Precision Calculations - Carl Rabe reports that the last lines of digits in several extended precision examples are incorrect, for example, on V12N2P24 the last ten digits of the square root of 3 should be 4587855397 not 1128740615. Similarly, on V11N2P25 the last ten digits of the logarithm of 2 should be 2344535347 not 2344535353, and the logarithm of 3 should be 2933973323 not 2933973343. Page 34 of this issue discusses the sources of these errors.

COLLECTOR'S CORNER - I continue to receive occasional requests for information about the older machines. For example, last summer a caller reported that he had purchased an SR-52 at a flea market for under five dollars. The device came with a few magnetic cards, but without manuals. The caller had learned to use the calculator, had loaded and run the programs from the magnetic cards, and had even written a few programs but had been unable to succesfully write a magnetic card. After some searching of my memory I was able to suggest that he needed to attach the self-adhesive magnetic tabs over the write-protect windows on the card. He didn't have the tabs, but called back within the hour to report that he had fabricated some tabs from other magnetic material, and had successfully written a program to a card! In the process of our telephone conversations I found that the caller was young. He had sort of taken it for granted that personal computers had been available for many, many years, and was bemused by my short oral history of the early programmable calculators.

Last month I received another call, this time from an individual who was interested in obtaining older calculators for his collection. He reported that he had 48 machines. I suppose that I qualify as collector of sorts. In the process of supporting the newsletter I have accumulated thirteen programmables, and have access to several more held by friends.

The caller asked if there was a club or organization for collectors. I don't know of one, but I know that former members Dave Leising and Maurice Swinnen have large collections. I also recall an announcement of an Hewlett-Packard display of some sort. If you are a collector and would be interested in contacting other collectors, or would like some help in locating an older machine for your collection, let me know. If there is enough interest we will start a collector's corner.

PAPER FOR THE PC-100 - Members continue to report difficulty in finding printer paper for the PC-100. I have a limited amount of paper available. Other members who no longer use their TI-59's may also have paper available. Write for specific details if you need help.

V11N4P26 reported that Labelon Thermal Calculator Paper Type CR-025 will work in the PC-100. Robert Lucas reported that the paper was available from L. E. Muran Co., 45 Dunham Road, Billerica, MA 01821. Telephone numbers are 617-667-4900 and 617-272-2208. I have not exercised that option. Can any member tell us whether that is a good source?

SUPPLIES FOR THE TI-74 AND TI-95 - One member has reported difficulty in obtaining accessories and supplies for the TI-95. I have found that the Service Merchandise outlets, the only support for those devices in the Tampa Bay area, had clearance sales on TI-74 and TI-95 material and no longer carry either equipment or supplies. The latest catalogs from EduCALC (Volume 41) and Elek-Tek (Volume 16) continue to list TI-74 and TI-95 material. The addresses and telephone numbers are:

Addresses:	EduCALC Mail Store 27953 Cabot Road Laguna Niguel CA 92677	Elek-Tek 6557 North Lincoln Avenue Chicago IL 60645-3986
Telephone:	(714)-582-2637	1-800-621-1269

Both firms will accept Master Card or VISA for telephone orders. If you order please mention our club.

I have successfully used Radio Shack's Thermal Paper for use with the PC-3 Printer (Radio Shack Catalog No. 26-3592) in my PC-324. A box of five rolls is priced at $2.49 plus tax. The paper seems to provide better contrast that the TI paper, but it's inconvenient to use since the rolls are so small -- each roll contains only 86 inches of paper. Both the TI paper and the Radio Shack paper are made in Japan. In V11N4P7 I commented on the high price of the PC-324 paper -- about 97 cents per cubic inch. The Radio Shack paper is only slightly less expensive, about 80 cents per cubic inch.

If you aren't on the distribution list for the EduCALC catalog you should be. TI-95 owners will be particularly interested in volume 41 of the catalog which contains a coupon for a free plug-in module. The offer is good through 12/31/88.

INTEREST SURVEY - I tabulated the reponses to the interest survey which was part of the subscription form for this year. For the first 95 subscribers the results were:

TI-59	-	59	TI-66	-	7
TI-95	-	54	fx-7000G	-	5
TI-74	-	30	HP-28	-	4
CC-40	-	10			

where the totals are greater than 95 because many members checked more than one machine. I also found that most members who checked the TI-95 had also checked the TI-59. One result of the response was my decision to include a compilation of PPX program availability in this issue.

FAST MODE - One new member writes "... I still don't have a clue as to why fast mode works, nor how it is done. Your current articles are undoubtedly quite clear and informative to older subscribers, but to me, and presumably to other recent subscribers, they are just as frustrating as arriving late for a movie, and trying to make sense of its second half."

A good introduction to the fast mode concept appeared in the July/August 1981 issue of PPX Exchange which is no longer in print. For a copy of the article send a couple of stamps to cover copying and postage.

V10N3P18-19 of TI PPC Notes contained a two page article on the origins of fast mode, including a demonstration program for printing the TI-59/PC-100 print code table. Add another stamp for a copy. Send an additional dollar if you want a magnetic card for the demonstration program.

In the past we offered a 36 page compilation of fast mode material from the 1980-1982 issues. We discontinued it because of complaints of legibility problems with some of the reduced size program listings. Generally, the text in the compilation was legible, and the time required to reassemble the program listings wasn't consistent with the infrequent demand for the compilation. If you want the compilation as is send five dollars.

The compilation primarily concerns the first method of fast mode entry, the 2nd Pgm 02 SBR 239 sequence starting at program location with the Master Library module installed. There has been no compilation for the issues from 1984 through 1987 where the implanted hexadecimal code (h12) and the Stflg at the end of the partitioning came into favor.

A SURVEYING MODULE FOR THE TI-74 AND TI-95 - The Survey-6 module for the TI-74 is an integrated coordinate geometry program designed for land survey calculations. It performs traverse, inverse, intersections and resection. The integrated program has the ability to store and recall coordinate pairs by your chosen name/number or automatic consecutive numbers. More than one coordinate pair with the same name, such as 'TANK' can be resident in memory because only points stored with the automatically attached current job code are normally visible.

Field staking calculations include stationed horizontal curve (offset from centerline, tangent, chord, secant), stationed vertical curve, stationed staight grade, vertically offset EDM reduction, and triangle solutions (sss, sas, asa, ssa). Printer output and cassette tape storage of named/numbered coordinates is also possible.

The program is currently available on a module for the TI-74 for ninety dollars. It expected to be available on a module for the TI-95 in the near future at the same price. For further information write to Ron W. Hardy, 2879 Lassen Ct., Las Vegas NV 89122, or call (702)-459-4312.

Editor's Note: This announcement does not constitute an endorsement by our club. It is provided for the possible use of members who are surveyors. If you decide to purchase the program please consider writing an evaluation for publication in a future issue.

ENTERING & AND % ON THE TI-95 - Don Laughery. TI-95 owners may or may not have received an insert with the TI-95 Programming Guide which concerned the entering of the & and % alpha characters. The insert stated:

> "If the calculator is the learn mode and you attempt to type either of the alpha characters & or %, the result is &+ or %5. You can obtain the desired character by editing after you type.
>
> To type & in an alpha message, press 2nd & <- . You can then type over or delete the extra character.
>
> To type % in an alpha message, press 2nd % <- . You can then type over or delete the extra character.
>
> Do not begin a label name with & or % unless you want the label name to be &+ or %5.
>
> Avoid using & or % in a file name. A file named with these characters outside the learn mode may be impossible to access from a program.

Experiments show that the &+ and %5 anomalies are key bounce problems of some sort. There is a way to enter the either & or % as the first character of a label without having to accept the + or 5 as the second character. The trick is to support the calculator in your hand clear of any support and tap the desired key lightly but quickly. This apparently works for the same reason reducing unsprung weight in a car design results in a better ride. Examples of results using the technique appear at the right.

```
0000 DFN F1: % @%%
0007 DFN F2: & @&&
0014 DFN F3:% %@&%
0021 DFN F4:&%&@%& HLT
0029 LBL %% 12345 HLT
0038 LBL && PI HLT
```

TRANSLATIONS FOR THE TI-59 - Scott Garver reports that he has completed translations of two TI-59 programs for use on the TI-95. The programs are: 628008A (Beams in Flexure) and 628124A (Built Up Sections). Scott has also performed other conversions. For information on availability send a stamped and self-addressed envelope to

Scott L. Garver
279 Central Ave., 2
Edison NJ 08817

I encourage others who have completed translations to make them available to other members of the club.

TI-88 WANTED - In mid-1982 TI announced that a new programmable calculator known as the TI-88 would be introduced later that year. The TI-88 was withdrawn before it became available to the public. However, a limited number of the devices were placed in the hands of selected users. If you know of a TI-88 which is for sale write to Michael Sperber, Birkenallee 67, 8526 Bubenreuth, Federal Republic of Germany. Old-timers will recognize Michael as one of the original developers of the high resolution graphics technique.

LISSAJOUS PATTERNS ON THE FX-7000G - Bob Patton. One of the annoying characteristics of the fx-7000G is that when a graphics program is running the display alternates between a text screen which contains any commands previously entered and a graphics screen containing the latest state of the curve. An example appeared with the random walk demonstration from V11N2P19. If you start with a clear screen and press Prog 1 EXE to start the program, then as the curve builds the words "Prog 1" flash in the upper left corner of the screen. When the first graph is complete, and you press EXE for a second graph the word "done" also flashes in the second line at the right side of the screen. When eight curves have been run a column of the word "done" will flash at the right side of the screen.

While this does no actual harm it is not pretty and watching the curves build is sometimes a large part of the fun. Ideally, there should be a command that can be put in a program to clear the text screen, but I have been unable to find one. The only command that does so from the keyboard is AC, but that command is not programmable. There are two solutions I know of. The clever one is to just put an alpha space, " ", in the graphics generation loop. This inserts one blank line per cycle and quickly clears the text as the program runs. Unfortunately this slows things down a little and still shows an annoying amount of flicker. The clean but brute force solution (it takes 16 steps) is a separate screen clearing loop that writes 8 blanks at the beginning of the graphics program. An example appears in the demonstration program below.

The program at the right will draw Lissajous figures. Steps 1 through 3 ask for input and define the dimensions of the display. The inputs in response to the prompts are:

J = number of vertical nodes, and

K = number of horizontal nodes.

Steps 5 through 7 are the screen clearing loop. The minus sign in line 8 is made with the Minus key not with the Subtraction key. Steps 11 through 17 are the main plotting loop. Step 18 leaves the figure in the display when the plot is complete. If you do not include step 18 the program stops with the text in the screen and it will be necessary to press G=T from the keyboard to see the completed figure.

The program will usually draw complete figures. J and K should be relatively prime, and normally J < K because of the shape of the display. However, try J = 1 and K = 2 versus J= 2 and K = 1. I haven't really tried to analyze that anomaly.

```
 1 "J"? → J
 2 "K"? → K
 3 Range 0,6,2,0,4,2
 4 8 → Q
 5 Lbl 1
 6 " "
 7 Dsz Q:Goto 1
 8 -180K → T
 9 20K÷1 → N
10 J÷K → M
11 Lbl 2
12 2.9sin MT → I
13 1.9cos T → J
14 Plot I÷3,J÷2
15 Line
16 T÷18 → T
17 Dsz N:Goto 2
18 Graph Y=0▲
```

Editor's Note: You might also try values for J and K which are not integers to get some interesting graphs; e.g., try J = 2.7 and K = 3.5.

RANDOM WALK ON THE fx-7000G - P. Hanson. V11N2P19 included a program for demonstrating random walk on a fx-7000G. That program only provided individual random walk plots. One of the attributes of the random walk phenomena is that the rms for a family of individual runs should approach a square root function as the number of runs becomes large. I modified the program from V11N2P19 to accumulate the sums of the squares and to plot the RMS on demand. The resulting program is at the right below.

```
 1  26 → A
 2  Lbl 5:0 → C[A]
 3  Dsz A:Goto 5
 4  Range -14,80,10,
    -32,31,20
 5  Lbl 3:Cls
 6  8 → A:80 → B:0 → C
 7  Lbl 2:" "
 8  Dsz A:Goto 2
 9  Lbl 1:A+1 → A:C-1 → C
10  Ran#<0.56 ⇒ C+2 → C
11  Plot A,2C
12  Frac (A÷4)=0 ⇒ C[A÷4]
    +C² → C[A÷4]
13  Dsz B:Goto 1
14  Z+1 → Z
15  Lbl 7
16  Graph Y=0
17  "1 = ANOTHER RUN"
18  "2 = PLOT RMS"
19  ? → Y
20  Y=1 ⇒ Goto 3
21  Cls:0 → B
22  Lbl 6:B+1 → B
23  Plot 4B,2√(C[B]-Z)
24  B<20 ⇒ Goto 6
25  Goto 7
```

Lines 1 through 3 which clear registers A through Z can be replaced with an Mcl command. These lines were necessary with my program because the Mcl function is not working in my fx-7000G.

In line 4 the range values are changed from those in the program on V11N2P19. If you had looked carefully at the plots from the original program you would have seen that there were some missing points; for example, for the thirteenth x value from the right hand side. You would also have seen that the scale markers on the axes were unevenly spaced, with some intervals having nine spaces and some having ten. The revised range values eliminate those problems.

Lines 7 and 8 clear the screen as described on page 6 of this issue.

Lines 9 through 13 generate the random walk plot. Line 12 sums the square of the plotted value for every fourth point. Note that an algebraic expression can be used as the subscript as indicated on page 125 of the Owner's Manual. The test value for the decision whether to move positive or negative is different from that in the original program to compensate for the change in the random number statistics as the loop length changes.

Line 14 accumulates the number of random walk plots generated. That value is used in the RMS calculation.

Lines 15 through 19 are a subroutine which freezes a completed plot in the display and provides two options for subsequent plots. To continue after a plot has been completed the user presses EXE. A prompt appears which allows him to select another random walk plot or an RMS plot. The user enters either a 1 or a 2 and presses EXE to select the option desired.

Lines 21 through 25 generate the RMS plot. Note that expressions are permissible for the dimensions of the plot command as illustrated in line 17 on page 171 of the User's Manual.

TI-66 WANTED - Will pay $25.00 for a TI-66 in operating condition. Ed Brewster, 15455 Glenoaks Blvd. #296, Sylmar, CA 91342. (813)-364-2278.

A BUG IN THE INV ASM FUNCTION FOR THE TI-95 - Robert Prins.

TI Holland gave me the name and address of a Mr. Petr Popov who was experiencing problems with the INV ASM function. Mr. Popov told me that his TI-95 would sometimes produce non-existent labels when using the INV ASM function. At first he wasn't able to produce the errors again, but later he phoned to tell me that he had found another error. It turned out that both his and my TI-95 produced the error.

This bug only occurs under certain conditions. In the case Mr. Popov discovered the conditions are:

Partitioning: File space has a size of 1680 bytes.

Steps involved: 1407 - 1413 containing DFN F1: W @B7

1422 - 1424 containing LBL B7

If you ASM a "program" with just those two instructions, the DFN will change into DFA F1: W @1425 which is perfectly OK. The problem occurs when you subsequently INV ASM this "program" as the DFA remains a DFA.

In the case of Mr. Popov's program the DFA was changed back into DFN F1: W @B: which was nonsense since his program doesn't even contain a LBL B: . Mr. Popov also found a similar bug in another program.

The tricky matter about this bug is that is does not occur when you insert a NOP between steps 1414 and 1422 or before step 1407. Altering the size of the internal file space also stops it from occurring.

Editor's Note: Robert has already forwarded a discussion of this bug to TI. I have had difficulty verifying the effects in some programs. In the meantime, users should be on the alert for problems when using the ASM and INV ASM functions.

SCROLLING DISPLAY - D. Laughery. Here is a little program which will flash the most recently entered alpha entry alternately with the last numeric display. Just enter the alpha message and run the program

LBL AA PAU OLD PAU CE GTL AA

As an example enter the alpha message

HI THERE, (15 spaces) DUMMY!

This will simulate a scrolling message sign.

USED TI-59'S - The club can continue to provide used TI-59's for fifty dollars plus shipping. Send sixty dollars. We will give fifty dollars to the present owner, ship the calculator to you and return anything remaining. You can save on shipping if you do not need the manuals. All of the units shipped will have passed the extended diagnostics in V12N1P4/5. Even so, all used units are provided entirely at the buyers risk.

MAILBAG

"I have an interest in converting my existing TI-59 programs to my new TI-95. I look forward to each new issue. You are doing a terrific job for us TI-95 users. Thank you very much." A.M.

See more discussion of translation in this issue.

"For future issues let's have more puzzles, extended precision programs, root finding, cuurve fitting and translations from the TI-59/TI-95 to BASIC." M.B.

"The newsletter gets better every year. I look forward to each issue." G.W.

"More on translation would be appreciated." J.A.

"Please publish a program list of all the current available programs, denoting what calculators they're written for with names and addresses of who to contact. It would seem to be a good idea to publish an updated list at least once each year." W.W.

Several years ago former member Bob Fruit published a comprehensive index of the material from the 1980 through 1983 issues. Among other things it included listings of programs by title and author. Unfortunately, there wasn't enough demand to cover the costs, and Bob discontinued the effort. If you would like a copy of the fifteen page index send three dollars to cover copying and postage. This issue includes a compilation of all of the PPX programs for the TI-59 which are available from our members.

"I enjoy the articles. Being the owner of a TI-74 I would like to see more articles on that calculator." P.T.

"... Have personally found it easier to forget TI-59 programs, study the TI-95 and rewrite programs without having to translate. The extended program space makes it possible to use the same subroutines with many programs." J.R.

"I'd like more BASIC language programs." R.B.

"I propose the following two very challenging problems for TI-95 programmers: perform symbolic math operations like that of the HP-28S, and solve polynomial equations without guessing initial roots like that of the 'Siljak Polynomial Root Finder' or discussed in Alexander Morgan's book Solving Polynomial Systems Using Continuation. Both references are advertised in recent EduCALC catalogs." P.M.

"Interested in the TI-66, TI-74 and Casio fx-7000G. I still use the TI-50." F.F.

"I am hoping someone will be able to convert the biorhythm TI-95 program for the TI-74." N.F.

See the converted program elsewhere in this issue.

"TI PPC Notes is as interesting to read as a love letter, and it has been a long, long time since I got a love letter. Keep up the good work!" W.H.

"Having a very hard time finding TI-95 accessories and printer paper. TI prefers not to compete with retailers, but nobody stocks support products." P.E.

The latest EduCALC and Elek-Tek catalogs continue to list TI-74 and TI-95 peripherals and supplies. See the discussion on page 3 of this issue.

ACCESS TO FORMER PPX PROGRAMS

In late 1984 our club set up an informal program exchange to preserve access to the TI-59 programs from TI's PPX Exchange. The first listing in V9N5 contained over six hundred programs. That number has grown to over 1100 programs through listings in subsequent issues. however, the listings have become inconvenient to use since they are spread through seven different issues. In response to suggestions from several members the following pages are a consolidated listing.

The programs are listed in order by PPX number with abbreviated titles. A code precedes each PPX number which identifies up to three owners of that program. For example, the code 12F which precedes the 108005 program for Small Business Accounting means that the program is held by the three members (or former members who have agreed to continue to provide service) as identified in the listing below.

The programs identified by code 1 are held by the club. I will loan these programs to members for one dollar per program (no checks, please, for amounts under five dollars). It is understood that the programs will be returned promptly to be available for use by other members.

Other members who are willing to provide programs prefer a variety of arrangements ranging from loan to providing copies for a fee. You should contact them for current status. Be sure to send a stamped and self-addressed envelope for their reply.

Code	Owner and Address
1	TI PPC Notes, P. O. Box 1421, Largo, FL 34649-1421
2	Maurice Swinnen, 9213 Lanham Severn Road, Lanham, MD 20706
3	Robert McQuattie, 5848 Cottonwood Drive, Lorain, OH 44053
4	Lem Matteson, 8313 Ward Parkway, Kansas City, MO 64114
5	Heinz Zuschlag, 180 Shelton Road, Trumbull, CT 06611
6	Bob Patton, 1713 Parkcrest Terrace, Arlington, TX 76012
7	A. E. Mackenzie, 160 Alto Drive, Oak View, CA 93022
8	Paul D. Sperry, 4260 Grinnell Avenue, Boulder, CO 80303
9	Jose M. Gallego, 2302 "D" Avenue, Apt. 104, P.O. Box 2746, National City, CA 92050
A	Harry Rosenberg, 1900 South Eads Street, Arlington, VA 22202
B	Myer Boland, 66 Overlook Way, Englishtown, NJ 07726
C	Robert Ericson, 32 Ferncrest Blvd., North Providence, RI 02911
D	Gilbert Farrior, 1277 North Watson Road, St. Louis, MO 63132
D*	means Gilbert Farrior also has the program on magnetic cards
E	Nicholas J. Manicone, Port Planning/Room 64 South, Port Authority of NY and NJ, 1 World Trade Center, NYC, NY 10048
F	Robert W. Mosely, 107 Hillsboro, Greenfield, TN 38230
G	Michael G. Kelley, 11002 W. 21st N, RR #9, Wichita, KS 67212
H	Thomas Wyamuller, Phoenix Mutual, 1 American Row, Hartford, CT 06115
J	Larry Parsons, 815 Atlantic Street NE, Warren, OH 44483
K	Albert Smith, 14280 Sandhurst St., Brooksville, FL 33573
L	Laurence Leeds, 10232 El Dorado Drive, Sun City, AZ 85351
M	Thomas Ceterski, 10010 Alderson St., Schofield, WI 54476
N	Val Barron, 90 Cedar Point Drive, Williams Bay, WI 53191
P	James Taylor, P. O. Box 174, Marblehead, MA 01945
Q	Morris E. Karp, 840 Threadneedle Street, No. 181, Houston, TX 77079-2820
R	Shiu Lum Lee, 1029 E. 102nd Street, Brooklyn, NY 11236

BUSINESS

9 018001 - Break-even Point Margin of Safety
9 018002 - DD633 Pricing Program
H 018003 - Monthly/Fiscal Year Linear Trend Projection
H 018004 - Organizational Tech. Level Index - Average Grade
2 028007 - Labor & Material Cost Estimate
1F 038001 - Inventory Report
F 038002 - Store Inventory Computation
9F 038004 - Inventory Control - EOQ
F 038008 - Inventory Tally File
19 048001 - Gompertz Growth Curve Fit
F 058001 - Employees Earnings
F 058002 - Time Card Addition
2AF 058003 - Small Business General Payroll
29 068002 - Webster Fixed Time Traffic Signal Delay
9 068003 - Vehicles Entering Intersection Green Light
C 068006 - Standard Queue Model M/M/C
8 068007 - Standard Queue Model M/M/1
C 068009 - Transit System Modelling
1 078004 - Life Insurance Cost
A 088005 - Loan Amortization
1 088006 - Anal. of Graduated Pay Mtg. Loan
9Q 088013 - Buying vs Renting
H 088014 - Maximum Allowable Monthly Rent
9 098002 - Invoice Preparation
9 098004 - Universal Multiple Discounter
F 098007 - Purchasing - Quantity/Price Break Analysis
1 098009 - Purchase Agreement Analysis
1 098010 - Sheet Metal Material Pricing
C 098012 - Advanced Cash Register
1 098015 - Cost Estimate for Construction

FINANCE

12F 108005 - Small Business Accounting
C 108007 - Straight Line/Declining Balance Depreciation
A 108010 - Sources and Uses
1F 128001 - Loan Analysis
Q 128002 - Constant Reduction Loan
1 128007 - APR for Prepaid Loan
9F 128011 - Skip Payment/Balloon Payment Loans
4 128013 - Yearly Amortization Schedule II
9 128021 - Loan Amorization
A 128023 - Omnibus Debt Amortization Schedules
4 128024 - Fast Mode Loan Payment Schedule
4 128025 - Statement Savings Daily Compounding ...
F 128026 - ATM Checking Account Random Reconciliation
Q 138001 - Auto Finance Contract
9Q 148002 - Household Budgeting
19 148003 - Budget Analysis
9 148005 - Long Distance Telephone Charge
AJQ 148007 - Checkbook Maintenance
4 148010 - Checkbook & Savings Account Balance
1Q 148013 - Checking Account Management
9 148020 - Discount/Interest
F 148021 - Consumer Price Index Conversion II
C 148024 - Electric Utility Billing II
G 158004 - PWAC Cost Study
129 178001 - Income Tax Form 1040

2Q 178003 - Federal Income Tax 1978
1 178005 - Tax Sheltered Retirement Analysis
1 178012 - Federal Income Tax
1Q 188003 - Universal Rate of Return
1Q 188004 - Call Option Ratio Writing
1Q 188009 - Call Option Spreading
1Q 188010 - Screen Stocks
1Q 188011 - Value of Call Option
Q 188012 - Universal Rate of Return
2 188014 - Stock Portfolio Record Keeping
AG 188023 - Market Bearer Stock Selection
A 188030 - Stock Option Straddle Evaluation
A 188042 - Searching for Call Option Spreads
A 188049 - Yields & Taxes: Bonds & Money Market Funds
1 188052 - Portfolio Monitor
A 188053 - Portfolio Analysis
1 188914 - Internal Rate of Return
CG 198012 - Net Cash Flow and Rate of Return
4 198047 - Savings Passbook Interest Calculations
9 198048 - Analysis of Financial Statements (Ratios)
9 198049 - Project Economic Analysis
6A 198058 - Financial Statement Analysis
A 198061 - Financial Statement Analysis
4 198069 - Days Interest
AG 198071 - Investment Portfolio Optimization

STATISTICS AND PROBABILITY

D* 208003 - Simple Regression, 4 variables
9D 208005 - Exponential Smoothing
2 208007 - Multiple Linear Regression
DHQ 208008 - Polynomial Regression (to 7th Power)
1 208009 - Two Variable Polynomial Curve Fit
D 208011 - Regression Analysis for Four Curve Types
BDQ 208013 - Best Fit to Eight Curves
D* 208014 - Testing Large Deviations
D 208015 - Comparison of Regression Lines by Covariance
A9D 208016 - Multiple Regression - 5 Ind. Variables
D* 208017 - Autocorrelation and Multicolinearity
D* 208018 - Prediction, Confidence Intervals, etc.
D* 208021 - Automatic Curve Choice - 6 Curves
8D* 208022 - Multiple Regression with Step-wise Option
D 208023 - Least Squares Parabola y = a + bx + cx^2
2 208027 - Time/Volume Learning Curve
D* 208028 - Multiple Linear Regression Models, etc.
D* 208029 - Multiple Regression - 9 Independent Variables
3 208033 - Sine Curve Fit
Q 208035 - Non-linear Regression
2Q 208039 - Logistic Curve Fit
13Q 208040 - Eight Curve Fit
29Q 208041 - Multiple Curve Fit
3FQ 208050 - Automatic Curve Fit (Nine equations)
Q 208051 - Evolutionary Curve Fitting
6 208056 - Rational Curve Fit
9 208057 - Data Fitting
9 208058 - Exponential Function Fit
12 208059 - Polynomial Curve Fit with Errors
9 208060 - Least Squares Polynomial Fit
89 208067 - Non-linear Regression Analysis
8 208076 - Least Squares Polynomial Fit

GP 398019 - Spherical Triangles
G 398025 - Vector Operations
4 398035 - Extended Range Factorials
L 398045 - Precise Multiplication
9 398049 - Rectangular-Spherical Coordinate Conversion
9 398050 - Ellipse & Circle Plotter
B 398054 - Precision Fraction to Decimal Conversion
9D• 398055 - 3 Dimensional Coordinate Rotation and Translation
3 398064 - Synthetic Division - Diminishing Roots
9 398065 - Cubic Spline Interpolation
P 398068 - Ordinary and Partial Derivatives
34 398070 - Roots of Quartic - Complex Coefficients
4 398084 - Quartic, Cubic, Quadratic Equations
K 398092 - Magic Squares
9 398094 - Polynomial Division
9Q 398096 - Polar Graphing Program
PQ 398098 - Spherical, Cylindrical, Rectangular Coordinates
F 398104 - Equation of a Circle
L 398107 - Monte Carlo Primality Test
H 398110 - Pythagorean Analysis with Integer Solutions
4 398116 - Prime Number Generator
L 398118 - Constant Arithmetic
9Q 398122 - Addition of Fractions
1B 398131 - High Speed Prime Tester
K 398132 - Pascal's Triangle Term Generator
7 398136 - Solid Geometry
7 398137 - Inscribed & Circumscribed Circles
14 398153 - Simultaneous Equations
B 398159 - Unlimited Precision Division
4B 398165 - Multiple Precision Square Root & Remainder
1 398171 - Exact Factorials to 610 Factorial
4 398172 - Quadratic & Linear Equations
3 398176 - 3-D Vector Calculator
K 398179 - Magic Squares, 3x3 through 6x6
B 398182 - The Multiplier
3 398184 - Compendium of Polynomials
B 398186 - Roots using Newton-Raphson Method
P 398190 - Arc Length, Surface of Revolution
3 398193 - Vector Rotation
4 398194 - Properties of Circles
K 398197 - Happy Numbers
3 398198 - Polynomial Change in Variable
K 398199 - Location of Primes of Specific Intervals
1 398201 - Prime Number Data Base Generator
1P 398205 - Simultaneous Equations
7 398210 - Line Circle Intersection
7 398212 - Solid Mensuration
7 398213 - Plane Mensuration
9 398220 - RPN Vector Calculator
1 398225 - Prime Factors of an Integer
C 398226 - Volume, Area and Circumference
4 398227 - Extended Scientific Notation
3P 398228 - Partial Derivatives
3 398232 - General Function for Numerical Analysis
49B 398239 - Triple Precision Arithmetic
3L 398242 - Square Root - Double Precision
3 398243 - Prime Factors of Integers
P 398247 - Circle Tangent to Three Other Random Circles
L 398250 - Natural Logarithms - Double Precision
P 398256 - Finite Differences

P 398264 - F(x) Any Delta-x with List and PLot
9 398266 - Arityhmetic with Fractions
4 398267 - Intersection of Two Chords
L 398270 - e^x (multiple precision)
G 398277 - Axis Rotation
14 398278 - 13 Digit Modulo 30 Speedy Factor Finder
1B 398280 - Muller Zeroes of Functions
2 398281 - Precision Division 24 to 120 Digits
7 398282 - Area Bounded by One or Two Curves
B 398283 - Precision Division (168 Digits)
34 398284 - Twenty Digit Arithmetic
C 398285 - Coordinate Rotation
3 398286 - High Accuracy Solution of Quartics
3 398287 - Transform Polynomials by Removal of Terms

NATURAL SCIENCES

2 408006 - Physical Constants
9 408007 - Mass of an X (naught) Particle
2 408012 - Physics Conversions
9 408013 - Gravity due to Latitude & Altitude
1Q 408015 - Specific Gravity Computation
2 408017 - Gravity Modelling 3D Dipping Prism
9 408018 - Rydberg Formula
4 408019 - Circular Orbits
9 408023 - Nuclear Binding Energy & Radius
9 408024 - Molecules in a Two Chamber Box
9 408025 - Radioactive Decay Series
9 408026 - Alpha Particle Scattering Ratio
3 408028 - Simple Pendulum
9 408032 - Linear/Rotational Kinematics
9 408034 - Cyclotron Equations
2 408036 - Ballistic Trajectory
1 408038 - Vibration Units Conversion
9 408040 - Density of Dry and Moist Air
9 408049 - Hydrogen-like Atomic Orbitals
P 408051 - Electric Field Potentials
9 408056 - Bohr Hydrogenic Atom
9 408065 - Refractive Index of Infrared Cells: II
9 408067 - Refractive Index of Infrared Cells: I
9 408071 - Electrical Potential for Two Point Charges
8G 408078 - Vector Shorts
12Q 418001 - Perfect Gas Law
D 418002 - Cubic Crystallography
2 418003 - Temperature Conversion
2 418004 - Arrhenius Temperature Acceleration
Q 418005 - Dilution
9Q 418007 - First Order Chemical Kinetics
Q 418008 - Second Order Chemical Kinetics - Type I
8Q 418009 - Second Order Chemical Kinetics - Type II
Q 418011 - Guggenheim Method of 1st Order Chemical Kinetics
9 418014 - Least Squares Activation Energy
19 418015 - General Thermodynamics
9 418016 - Aqueous Acid/Base Buffer Equilibria
1 418017 - Acid-Base Titration
9 418025 - Radial Distribution Function (Quantum)
9 418026 - Mole Fractions of a 2 Component System
9 418027 - Composition of a 3 Component System
9 419028 - Surface Tension by the Ring Method
9 419029 - Conductance of Strong & Weak Electrolytes

1 418031 - pH - 1
1 418032 - pH - 2
1 418033 - pH - 3
1 418034 - Nernst Equation
Q 418037 - Periodic Table of the Elements
9 418038 - Molecular Parameters of a Diatomic Molecule
9Q 418039 - Conversions of Concentrations of Solutions
Q 418042 - Temperature Correction for Equilibrium Solution
9 418044 - Partial Molal Volumes ...
9 418045 - Viscometry
1Q 418048 - Mole, Weight & Volume Conversions for Gases
Q 418049 - Concentration Conversion
9 418052 - Fractional Distillation
9 418053 - Debye-Huckel Equation
19 418054 - Vapor Pressure & Heat of Vaporization
9 418055 - Viscosity of Water
9 418056 - Colligitive Properties
9 418057 - Potentiometric Titration
9 418063 - Standard Potential of Metal Electrodes
9 418065 - Gaseous Diffusion
9 418066 - Hueckel Molecular Orbital Method
Q 418067 - Formula and Composition (Replaces 418006)
Q 418068 - Concentration Exchange
8 418069 - Half Life
1 418070 - Intrinsic Melt Viscosity
9 418071 - Consecutive Reversible Kinetics
9 418073 - Fixed Points/Phase Equilibrium Boundaries
9 418074 - Root Mol. Wt. Determination
1 418076 - Alkyl/Polyester Theoretical Mol. Wt.
9 418084 - Quantitative Analysis by Mass Spectrometry
4 418090 - Periodic Table 107
9 418097 - Thermodynamic Parameters
9 418109 - Values of the Gas Constant
8 418115 - Ph of Mono-, Di-, Triprotic Acids and Buffers
9 418117 - Water III - Vapor Pressure
9 418119 - Water II - Heat of Vaporization
9 418122 - H-like Atomic Orbitals II: Contours
9 418123 - Water V - Liquid Specific Volume
9 418124 - Water IV - Steam Volume
9 418125 - Water VII - Steam Viscosity
9 418126 - Water VIII - Heat, Free Energy
9 418127 - Water VI - Heat Conductivity
38 418131 - Balancing Chemical Equations
4 418132 - Complete Electronic Structure of Atoms
2 468001 - Magnetic Modelling 2D Prism
2 468005 - Mineral-Field Check
8 498002 - Skew/T Log-P Adiabatic Chart

LIFE SCIENCES

12Q 508001 - Calorie Determined Weight Program
4 508013 - Serum Ethonal Level (Drunk Test)
1 548007 - Scanning DNA - Enzyme Sites
9 588003 - Custom Metric Diet Planner
G 588006 - Diet Calculations for Kidney Patients

ENGINEERING

8 608002 - Sattleite Finder
8 608007 - Approximation of the 1962 Standard Atmosphere

G 608010 - Rib and Spar Intersection at Datum
18 608017 - U.S. Standard Atmosphere - 1976
G 608020 - Airplane Weight Engineering Program
1Q 618001 - Flash Calculation - L/V Ratio to be Determined
1Q 618002 - Redlich-Kwong Equation
1Q 618003 - Bubble and Dew Point Temperatures
Q 618004 - Hydraulic Loop Calculations (Flow)
29Q 618005 - Beattie-Bridgman Equation of State
1Q 618008 - Solution to Pipe Problems - Flow
1Q 618009 - McCabe-Thiele Binary Distillation
12Q 618010 - Steam Tables - Vapor
1Q 618011 - Gas or Steam Piping Flow Pressure Drop
1Q 618012 - Heat Transfer Coef/Pressure Drop
1Q 618013 - Incompressible Fluid Pipeline Pressure Drop
1Q 618014 - Heat Exchanger Thermodynamics
1Q 618015 - Heat Exchanger LMDT
1Q 618017 - Orifice Calculations for Liquids
Q 618018 - Binary Distillation
Q 618020 - Valve Sizing for Perfect Gas Flow
Q 618021 - Valve Sizing for Liquid Flow
Q 618022 - Valve Sizing for Saturated Steam Flow
Q 618024 - Gas Orifice Calculations (Flange Taps)
1Q 618029 - Fluid Flow - One Phase
Q 618031 - Mean Temperature Difference Exchangers in Series
Q 618032 - Multicomponent Enthalpy Calculations
Q 618034 - Average Particle Size
Q 618035 - Heat of Reaction and Rate Constant
1Q 618036 - Two PHase Flow Resistance
Q 618037 - Relative Humidity/Dew POint
Q 618038 - Pipeline Gas Velocity
Q 618040 - Heat Capacity and Enthalpy
Q 618042 - Slurry (Specific Gravity/Dry Solid Rate)
Q 618044 - Steam Flow Rate throgh Orifice Plate
Q 618047 - Pipeline Pressure Drop - Compressible
1 618051 - Restrictive Orifice Sizing
9 618058 - Critical Properties of Organic Substances
8 618071 - Peng-Robinson Equation of State
1GQ 628001 - Trapezoidal Channel Depth and Velocity
1G 628002 - Four Span Moment Distribution
1GM 628003 - Moment of Inertia
1 628004 - Warren Truss Solution
1 628005 - Dynamic Loading/Single Degree of Freedom
1 628006 - Concrete Beam Stress Analysis
R 628008 - Beams in Flexure
M 628010 - Simple Beam with two Cantilever Ends
1 628011 - Reservoir Design
E 628013 - Lateral Earth Pressure Coefficients
R 628016 - Bearing Plate Design
R 628017 - Beams in Flexure with PC-100A Labels
ER 628020 - Moment Distribution: Variable Number of Spans
R 628021 - Base Plate Design
1Q 628022 - Hazen-Williams Formula (Pipes)
1Q 628024 - Equivalent Pipe Method
R 628025 - Wall Footing Design
Q 628028 - Pipe Flow Calculations
M 628032 - Circular Concrete Column
M 628035 - Rectangular Concrete Column Analysis
M 628036 - Masonry Wall Design
Q 628038 - Head Loss over a Submerged Weir
Q 628040 - Flow through a Circular Pipe

2 658015 - Radar Range Equation
2 658018 - Active Band Pass Filter
2 658019 - Active High Pass Filter
2 658021 - Active Low Pass Filter
2 658023 - Active Low, High and Band Pass Filters
2 658024 - Low Frequency Transistor Amplifier
P 658027 - Resistor Triangle Measurements
1 658029 - Ohms and Power Law
FN 658030 - Distance by Geographic Coordinates
2 658034 - Coax Data
2 658039 - 1% Resistor Table
F 658040 - Third Order Intermodulation Products
F* 658041 - T or Delta Attenuation Design (1 card)
2 658042 - Wien Bridge Notch Filter
N 658045 - Ground Station G/T
29 658048 - Graphical Fourier Analysis
F* 658053 - Electronic Conversion (1 card)
1 658054 - Amps, HP, KW and KVA
1 658055 - Transformer Analysis
1 658060 - Volts, Watts, db, etc.
2 658070 - Speaker Crossover
2 658078 - Noise Prediction
2 658089 - Array Antenna Beam Calculations
2 658092 - Atmospheric Ray Trace (Radar)
8 658094 - Millivolts to Temperature
F 658095 - Business Band Range
F 658105 - Chebychev Active Lowpass Filter & Pole Locations
F* 658108 - The Compleat Attenuator (2 cards)
Q 658111 - Resistor Value Decoder/Encoder
2 658112 - TV Channels
N 658115 - Digital Circuit Truth Table Generation
2 658118 - Ant. Log-periodic Dipole Array
P 658129 - Heat Sinks for Semi-conductors
N 658136 - Antenna Heights Required for L.O.S. Radio Path
N 658140 - Microwave Path Profile Calculations
N 658150 - Azimuths, Path Length & Coordinates of Intermediate Points for Radio Path
P 658159 - Inverse LaPlace Transforms
N 658160 - Microwave Path and Parabolic Antenna Parameters
N 658161 - Microwave Beam Clearance & Parabolic Antenna Parameters
N 658162 - Over the Water Microwave Path Reflection Points
N 658165 - F.M. Microwave System Noise
N 658166 - 4 and 6 GHZ Frequency Diversity Combinations
N 658167 - Unavailability of Microwave Paths due to Multipath ...
2F 658173 - Potentiometer Design
2F 658175 - Bipolar Transistor Design - Current Mode
F 658188 - Log-periodic Dipole Array Design
Q 668001 - Pipe Calculations
1Q 668003 - Fan/Blower Ratings
1 668005 - Welded Branch Reinforcement
2 668006 - Drill Size number Table
2 668007 - Hypodermic Tubing
9Q 668009 - Psychrometric Calculator (Air Properties)
1Q 668011 - Weymouth Gas Pipeline Pressure Drop
1Q 668013 - Centrifugal Compressor
1Q 668015 - Gas or Liquid Pipe Size
1Q 668017 - Flue Gas Enthalpy
1Q 668018 - Steam Flow through Pipe
P 668020 - Rectangular/Delta Strain Gages
8 668024 - Plane Stress Analysis

1Q 668030 - PV-01 Pressure Vessels
P 668031 - One Dimension Heat Conduction
1 668032 - Combustion Parameters
7Q 668033 - PV-02 Pressure Vessels
7Q 668041 - PV-03 Pressure Vessels
7Q 668042 - PV-06 Pressure Vessels
7 668044 - PV-05 Pressure Vessels
7Q 668049 - PV-04 Pressure Vessels
7Q 668053 - PV-07 Pressure Vessels
Q 668056 - Nozzle Calculation
1 668059 - Mass Moment of Inertia
1 668060 - Hydraulic Complex Impedence in Pipe
1Q 668061 - Heat Transfer through Insulated Pipe
2 668073 - Taper Bore Check
18P 668075 - 3-D Stress Analysis (Mohr's Circle)
P 668076 - 2-D Mohr's Circle
P 668079 - Section Properties
2 668084 - Dovetail Check with Rolls
1 668088 - Anemometer & Pitot Tube Calculations
1 668090 - Air Fuel Ratio - Automotive
1 668092 - Heat Transfer, Air Streams
7 668105 - PV-09 Pressure Vessels
Q 668129 - Relief Valve Sizing
1Q 668130 - Safety-Pressure Relief Valve Sizing
Q 668131 - Pressure Relief Valve Sizing
P 668149 - Rectangular Fin Thermal Analysis
1 668168 - Restrictive Orifice Sizing
P 668174 - Maximum Stress - Rectangular Flat Panel
G 668193 - Fed Std H28 Thread Dimensions
9 678001 - Critical Mass
2 688005 - Bit Error Probability (Transmission)
1 688006 - Air Conveying
P 688007 - Second Order Underdamped System Response
Q 698001 - Perspective Drawing
1 698002 - Dessicant Dryer Sizing
12Q 698004 - Perspective 3d Option, Illustrator's Aid
9GQ 698006 - Axonometric Projection
9Q 698007 - Isometric Projection
9Q 698008 - Oblique Projection
P 698014 - 45 Degree Rosette Analysis- (Stress)
8 698028 - Sheet & Wire Coils
C 698030 - Fire Sprinkler System Design
P 698036 - Loads due to Pitch and Roll of Ship

TECHNICAL

C 718003 - Illumination at a Point in Interior Space
C 718006 - Lighting Calculations (Watts/Square Foot Method)
C 718010 - Lighting Calculatons (Zonal Cavity Method)
1Q 738001 - Heating Load Calculation
1Q 738002 - Heating and Cooling CFM Distribution
1 738003 - Air Conditioning Basic
1F 738005 - Residential Heating Load
1Q 738006 - Duct Design II
C 738015 - Air Conditioning and Heating System
C 738020 - Solar Gain Analysis
B 740015 - Meridional Ray Trace
1 748001 - Software Development Estimator
B 748007 - Nakautov Telescope - Skew Rays
B 748021 - Ray Trace - Newtonian

B 748024 - Optical Ray Trace - Meridional
B 748031 - Optics - Third Order
Q 758004 - I'll Do It Myself (Programming)
GQ 758005 - BASIC Language
2 768002 - P & S Seismic Wave Velocities
F• 778017 - Field Area Measurement (1 card)
G 778042 - Bearing Rotation
C 778043 - Hasty Survey
9 788003 - Mean Sidereal Time
9K 788015 - Moon Phase for Any Date
2 788025 - Hohmann Transfer Orbits
F 788031 - Geostationary Satellite Locator
9 788045 - Planet Finder
2 798017 - Draftsmen's R/A Trigonometry
4 798021 - Add and Subtract Feet and Inches
4 798029 - Horizontal Sun Dial
P 798047 - Developed Length of Tubes
P 798057 - Center Finder for CMM
G 798061 - Printing Price "20"
4G 798063 - Perspective Drawings of Objects

SOCIAL AND BEHAVIORAL SCIENCES

9 838001 - Manpower Utility Model
9 838003 - Skid Mark Evidence
1 858003 - Max Economy, Multiple Effect Evaporators
Q 868001 - Well Log Interpretation
Q 868002 - Rapid Gas Reserve Estimate
1Q 868003 - Compressibility Factors for Sweet Natural Gases
Q 868004 - Production Schedule for Exponentially Declining Wells
Q 868005 - Gas Well Deliverability
Q 868006 - Oil Reserve Estimate Constant Percentage Decline
Q 868011 - Gas Flow Measurement through an Orifice Meter
Q 868012 - Accumulator Sizing

GENERAL

1 908002 - Number Storage and Retrieval
12Q 908004 - Alpha Printing Clock
19Q 908005 - Flag Tester
17Q 908006 - RPN Simulator
1 908009 - Cartesian Graph
2 908010 - SR-52 Program Listing
12Q 908012 - Function Plotter
12Q 908013 - Bar Graph Plotter
1 908014 - Alphanumeric Register Listing
19Q 908015 - TI-59 Banner Program
169 908016 - Memo Pad
9K 908017 - Calendar Print Generator
1DQ 908018 - Universal Plotter
DQ 908019 - Multiple Plot
2 908021 - Cheating Test
2Q 908022 - Utility Routines II: Register Manipulations
23Q 908023 - Utility Routines I: Indicators
13Q 908024 - Utility Routines III: Display
Q 908025 - Print Conversion Integer Form
1Q 908026 - Conversion Routine: Numerals to Alpha Code
5 908027 - Store/Sort Utility Program
24 908030 - Sort/Merge (Shell-Metzner)
29Q 908031 - Data List

Q 908033 - Plotter/Printer
9 908035 - Table Summation
29Q 908036 - Extended Data Plotting
DQ 908038 - Functions on Memory
5Q 908039 - File Management
K 908042 - Add and Subtract Time
1 908044 - Linear Sort & Matrix Sort
6KQ 908045 - Large Alphanumerics
D 908046 - Slides and Merry-Go-Rounds
C 908047 - Expanded Memory
1FQ 908050 - Index Numbers - Store and Recall
1Q 908051 - Print Conversion (with Decimal and Sign)
D 908054 - Utility Grid
2Q 908055 - Op 07 Extended
1DQ 908056 - Utility Routines IV: Flags
DQ 908057 - Utility Routines V: Miscellaneous
DQ 908059 - Utility Routines VII - Register Consolidation
DQ 908060 - Utility Routines VIII: Registers
2D• 908061 - Program Relocator
56Q 908062 - Files
12Q 908063 - Hierarchy Register Functions
DQ 908064 - Bar Graph Printer with Labels
Q 908065 - Mini LIst/Message Composer
Q 908066 - Constant Significant Figure Engineering Format
Q 908067 - Right Margin Numerical Printer
Q 908068 - SR-52 Program Listing
1Q 908069 - Alphanumeric Register Listing with Tags
K 908073 - Cryptographic Encoder-Decoder
4 908077 - Master Conversions Length
4 908078 - Master Conversions Volume
4 908079 - Master Conversions Weight
4 908080 - Master Conversions Fluid Capacity
4 908081 - Master Conversions Area
DFQ 908082 - TI-59 Memory Structure
4 908083 - Ballberger (97) Sort
2 908084 - Co-ord. Ordering Porgrams (Sort)
9DQ 908090 - Multiple Strip Print/Plot
Q 908092 - Print Format Routine
DQ 908095 - Data Register Manipulations
2Q 908097 - RPN using HIR
16Q 908099 - Indirect Address Labeling
9Q 908101 - Histogram, Bar Graph, Point Plotter
Q 908102 - Cartesian Graph Conversion
1 908104 - Stop Watch Timer
19Q 908105 - Improved Cartesian Graph
9Q 908107 - Data Packing & Unpacking
4 908108 - Numeric Sort
5 908109 - Sum Columns & Rows Simultaneously
2Q 908114 - Group Register Operations
HKQ 908115 - Utility Routine: Subroutine and Loop Timer
1Q 908116 - Alpha Printout, System Order Status
2Q 908117 - Alphabetic Register Loading
4Q 908118 - SR-52 Program Converter
12Q 908119 - Magnetic Card Comparator
12Q 908120 - Utility - Conditional Transfers
Q 908121 - Plttter, Op 07 with Background Grid
Q 908123 - Integer Labeling Subroutine
KQ 908125 - Message Encoder/Decoder
1Q 908127 - Alphanumeric List
Q 908128 - Variable Print Symbol Banner

26 908129 - Alpha Register Print
Q 908130 - Vertical Banner Program
2 908131 - Full 13 Digit Printer
9F• 908132 - Telephone Number - Equivalent Name
14Q 908133 - Seven Letters from Display Register
FQ 908134 - Word Processor
Q 908137 - Coordinate Graphics
3 908138 - Irregularly Spaced Data Plot
F• 908140 - Superplotter (1 card)
1H 908142 - Bucket Sort (up to 99 items)
39 908143 - Plot F(x,y)
13 908145 - Op 7 Extended
5 908147 - General Utility Plot
C 908150 - Sequential Operations, Data Array
9 908151 - Data Encryption
3 908152 - Utility - Op 10
6 908153 - Multiple Indirect Levels (Memory)
K 908160 - Even More Alphanumeric Register Listers
4 908162 - SR-52 Listing Mark 3
1 908164 - Writing & Alphabetizing Words
9 908170 - M/U Superplotter
1J 908172 - Character Set Printout
1 908175 - Memory Malfunction Diagnostic
Q 908181 - SR-52 Program Converter
PQ 908182 - Advanced Use of Registers
P 908183 - Address Labels
9 908184 - RPN-59
1K 908192 - High Speed Calendar Printer
9 908194 - Dual Data Column Print
K 908199 - Headline Printer - Vertical and Horizontal
9 908200 - Triangular Diagrams
Q 908201 - Indirect Addressing with HIR Registers
4 908204 - Reversible Banner
C 908210 - Utility Charges
13P 908211 - Exploit the T Register
P 908215 - Simplified Triangles
K 908219 - Mini-banner Program
4K 908226 - List 13 Digit Registers Fast
K 908229 - Complete Date Print
9 908231 - Working Days to Calendar Days
1J 908232 - Typewriter
1JK 908241 - Correcting Typewriter
1 908904 - Computer Art (LE-19)
2 918002 - Monopoly Banker
Q 918004 - Supersleuth
Q 918008 - Star Buster
Q 918009 - Learning Nim
K 918010 - Phantom Ship
Q 918012 - Baseball
Q 918013 - Parachute
K 918014 - Football
Q 918015 - Yahtzee
24Q 918016 - Tic Tac Toe
KQ 918018 - Golf
K 918019 - Princeps Puzzle
19J 918020 - Skydiving
24Q 918021 - Craps Revisited
79 918023 - Stocks & Bonds Simulation
29Q 918024 - 3D Tic Tac Toe
249 918025 - Checkers

K 918026 - One Arm Bandit
K 918027 - Card Dealer -- Bingo Caller
KQ 918028 - Football (Solitaire)
Q 918029 - Time Bombs
14Q 918030 - Poem Machine
K 918031 - Super Code Breaker
5K 918032 - Game of Gale
K 918035 - Scrambled Eggs
Q 918036 - Triple Yahtzee Scorekeeper
KQ 918037 - Electronic Golf
KQ 918038 - Horse Race
Q 918040 - Offensive versus Defensive Football
4K 918043 - The One Armed Bandit
2 918044 - Cashier
7 918045 - Cycling Biorhythm Printer with Calendar
7 918046 - Biorhythm Calendar
Q 918047 - Space War
4KQ 918048 - 7 Card Low Poker
9Q 918049 - Hexpawn
K 918050 - Keno
9Q 918051 - 31 - A Card Game
5KQ 918056 - Automatic Blackjack with Options
KQ 918060 - Hamurabi
29K 918061 - Pinball
K 918062 - Calculator Ping-pong
K 918064 - Roulette Challenge
K 918065 - Game of Kaylee
KQ 918066 - Basketball
K 918068 - Evens Game
K 918070 - Arithmetic Game
9K 918071 - Memory Flashcard
K 918072 - Intelligence Tester
4KQ 918073 - Automatic Draw Poker
K 918074 - Game of Qubic
KQ 918075 - Racetrack
K 918076 - Game of Life
Q 918077 - Mars Lander with Display
KQ 918079 - Deluxe Baseball
KQ 918080 - Ant Invasion
KQ 918081 - Game of Blitz
K 918082 - Double Play Game
KQ 918083 - Battleship
9K 918084 - Super Mindbreaker
Q 918085 - Astrogator Game
K 918086 - Simon, 2 Digits
K 918087 - Simon, 3 Digits
K 918088 - Aircraft Carrier Landing
4KQ 918090 - Seven Card Stud Poker
Q 918094 - Daily Biorhythm Compatibility
K 918095 - Casino Roulette
Q 918096 - Wumpus
9KQ 918097 - The Black Box
Q 918099 - Space Flight
KQ 918100 - Blackjack for 13 Players
1KQ 918101 - TI-59/PC-100 Demonstration
K 918102 - Hi-9
K 918103 - Interchange
K 918104 - Number Guesser (You Pick It, I'll Find It)
2K 918106 - The Knight's Tour
4KQ 918107 - Automatic Crap Game

O 918108 - Klingon Space Attack
K 918109 - Pattern
K 918110 - Peg Jump
K 918111 - Multiple-player Nim
K 918113 - One Dimensional Life
KQ 918115 - Hangman
9Q 918116 - Star TRek (Mini-version)
4KQ 918117 - Action Craps
KQ 918118 - Anvanced Baseball
K 918119 - Chuck-a-luck Dice Game
KQ 918120 - Pattern PLanner Game
KQ 918121 - Dungeons and Dragons
K 918122 - Hanoi Game Simulator
KQ 918124 - Wipeout!
KQ 918126 - Duel
K 918127 - Wythoff's Nim
Q 918128 - Expert Wumpus
K 918129 - Bullseye Game
KQ 918132 - Road Race
2K 918133 - Stars - Number Guessing Game
K 918134 - Eight Calculating Queens
K 918136 - Blackjack
K 918140 - Tic Tac Toe
2K 918142 - Son of Jive Turkey
K 918143 - Magic Square
K 918145 - 999 Game
KQ 918146 - Battleship
47 918148 - Roulette-59
K 918149 - Jumbled Letters
K 918150 - Bango
KQ 918151 - Hangman
49 918152 - X-country Auto Race
K 918154 - Solitaire Checkers
K 918158 - Greedy
K 918161 - Space Search
KQ 918162 - Cross Country Auto Race
1KQ 918163 - Hunt the Wumpus
KQ 918164 - Multi-player Race Car
K 918166 - Rock, Scissors, Paper
1 918167 - The Remainder Game
2 918168 - Nim with Strategy
4 918169 - Checkers II
K 918172 - Ski Race
Q 918174 - Space War (2 Dimensions)
Q 918176 - Tank Battle for Two Players
9Q 918178 - Star Wars
1 918182 - Dungeon Master's Aid I
9KQ 918185 - Baseball III
4K 918186 - Improved Tic Tac Toe
K 918187 - Checkmate!
K 918188 - Roll Five Dice
K 919189 - Two Dice Simulator
K 918190 - N5ch60ach4a' P4zz3e
K 918191 - Trap Game
K 918192 - High Stakes
4K 918193 - Poems with Six Letter Words
47 918198 - Las Vegas Blackjack
K 918199 - Bingo Card Printer
K 918200 - Chomp
4 918202 - Playing Card Shuffle - Draw & List

K 918203 - Poems with Seven Letter Words
K 918204 - Phantom Banner in Own Background
K 918207 - Pascal Triangle Art
K 918209 - Russian Roulette
K 918211 - Baccarat
K 918212 - Thaipan
K 918213 - Pick a Number
K 918215 - The Mansion
125 918217 - Backgammon
9K 918218 - 4-D Lunar Lander
K 918219 - The Joker's Wild
K 918221 - Chase
K 918224 - Hexapawn
K 918225 - Cribbage Showing
K 918226 - Anagrams
K 918228 - Pinball II
K 918229 - Othello
K 918230 - The Dragon
K 918231 - Oasis
K 918232 - What Now
K 918233 - Treasure Hunt
K 918237 - So Sorry
K 918238 - Asteroid Attack
K 918239 - Anagrams
K 918241 - Hangman Illustrated
K 918242 - Presidential Campaign
9K 918245 - Misadventure
K 918246 - Ballistics Game
K 918247 - Chemin de Fer
K 918248 - Le Cave
K 918249 - San Francisco Liar's Dice
K 918251 - Awari
K 918252 - Odd Ball
K 918255 - Mouse in a Maze
K 918256 - Buried Treasure
K 918257 - 3x3 through 6x6 Magic Square Generator
K 918259 - Hunt the Mugwump
K 918260 - Reverse
K 918261 - Adventurer
4 918262 - 3-D Mars Lander
K 918264 - Barbudi (Bar Buddy)
K 918265 - Revised H-Lo
K 918266 - Othello II
7K 918267 - Las Vegas Blackjack #2
9K 918268 - Bingo
4K 918269 - Tic Tac Toe for Two
K 918270 - Instant Insanity
K 918271 - Rescue Mission: Planet Nari
K 918272 - Buck Rogers (with AM Radio Tiki Sounds)
K 918273 - Alien Invasion (with AM Radio Sound)
K 918275 - Baccarat
K 918277 - Simon Says
K 918278 - Drag Race
K 918279 - Monopoly (R) Game (for printer)
K 918280 - TI-59 Demonstration (Deluxe Version)
K 918281 - Yot Race
K 918282 - Jive Turkey Revisited
K 918283 - Darts
1JK 918286 - Rule Away
K 918287 - Cover Up

K 918288 - Elongated Banner
1JK 918290 - Poem Machine II
K 918291 - Santa Claus and His Reindeer
3 918294 - Chess Description Notation
K 918295 - Snakes and Ladders
K 918296 - Two Deck Shuffle (Mixed)
K 918299 - Ice Hockey Game
K 918300 - Pictorial Craps
K 918301 - Liars Dice
K 918305 - Computer Perfection (R)
K 918306 - Towers of Hanoi Game with Display of Towers
K 918307 - Richelieu Valley Golf
K 918310 - Farkler
4 918311 - Biorhythms Double Critical Days (Fast Mode)
4 918312 - Biorhythms All Critical Days (Fast Mode)
K 918314 - Recursive Tower of Hanoi
9 918315 - Interactive Battleships
K 918318 - Cokes
9 928006 - Math Tutor
K 928008 - Interactive Arithmetic Teacher
K 928022 - Spell-a-number
K 928033 - Sum of Digits Game
7 928040 - Exploring Linear Equations
K 928041 - Sum of Three Digits
7 928054 - Equations (1)
7 928056 - Equations (3)
7 928057 - Equations (5)
7 928059 - Equations (2)
F 928148 - Plot of F(x,y)
8 938009 - Radial Distance or Position from a Station
K 938015 - Distance between Two Geographic Points
K 938017 - Distance by Rudge's Formula
8 938034 - Vortac Navigation with Winds
G 938041 - Position from Two VORTACS
4 938911 - Great Circle Flying (AV-11)
7 948001 - Compass Adjustment/Sun Azimuth
7 948002 - Height of Tide
79 948003 - Tidal Current Prediction
7 948006 - Mercator Sailing
29 948014 - Sunrise/Sunset
7 948020 - Great Circle Marine Navigation
7 948023 - 3 Star Fix
4 948028 - Marine Conversions
7 948029 - Sight Reduction Tables
7 948031 - Celestial Running Fix
7 948033 - Ships DR Navigation from Two Fixes
P 968014 - Music Transposer
Q 968016 - Single Song/Album Sorter
1 988004 - Individual Bowling Average and Handicap
1 988009 - Continuous Bowling Match Results & Projections
F 988013 - Continuous Readout, Real Time Road Rally Ave Speed
P 988058 - Sailboat Race Navigation
2 998004 - Date of Easter Sunday
F 998028 - Automobile owning/operating costs
K 998042 - Julian Calendar Adjustment for ML-20
K 998043 - Date in the Future
4K 998094 - Julian Day Number Calendar

Available for loan from TI PPC Notes

STATISTICAL TESTING

218001 - Chi Square, 2 Way Anova
218002 - Randomized Block Design - Anove Test
218004 - 2-Way Anove without Replication
218005 - One Way Analysis of Variance
218006 - Two Way Anova with Interactions
228001 - Rho (RS): The Spearman Rank Correlation Coefficient
228002 - 3x3 Covariance Matrix and Correlation Coefficients
228003 - Multiple Partial and Linear Correlation Coefficients (3 Variables)
228004 - Mann-Whitney-Wilcoxon Test/Compering Distributions
228005 - Nonparametric Correlations (Spearman Kendall)
228006 - Wilcoxon/Mann-Whitney Two Sample Test

LAB CHEMISTRY

418001 - Perfect Gas Law
418006 - Elemental Composition
418014 - Least Squares Activation Energy
418015 - General Thermodynamics
418016 - Aqueous Acid/Base Buffer Equilibria
668009 - Psychrometric Calculator

PROGRAMMING AIDS

638003 - EBCDIC Code Converter
638004 - ASCII Code Converter
638005 - ASCII and EBCDIC Encoder
638006 - TI-Programmer Simulator
638007 - TMS 9900 Disassembler
638008 - INTEL 8080 Disassembler

BLACKBODY

748001 - Blackbody Photon Radiance
748002 - Spectral Responsivity
748003 - Blackbody Energy Radiance
748004 - Detectivity (D*) and Responsivity (R)
748005 - Blackbody Flux (Signal)

PRINTER UTILITY

908004 - Alpha Printing Clock
908005 - Flag Tester
908009 - Cartesian Graph
908012 - Function Plotter
908013 - Bar Graph Plotter
908014 - Alphanumeric Register Listing
908015 - TI-59 Banner Program
908016 - Memo Pad

Data entry - Robert AH Prins The following article describes a few of the many methods to enter data into a TI-95 program. Although this may seem a rather trivial task, it gets a bit more complicated if you want your programs to be user-friendly.

When discussing data entry, we find that there are basically three types of programs:

- those that need only one input value
- programs that need up to about ten, usually "unrelated", input values
- programs that need more than ten, usually related, input values

Programs requiring one input value

If your program needs only one input value, it hardly seems necessary to write a special input routine, because you can put the input in the display and press *RUN <PGM>* or *RUN <DIR> <FIL>*. Although this method is very simple, I would recommend you use it only for programs under development and never for programs that you want to save in File Space [=FS]. Why? Well try to run this kind of program several times when it is the 50th file in FS. Each time you will have to press *RUN <MEM> F5 (12x) <FIL>*.

If you really have programs which require only one input value, it is a much better method to start the program and let it give you a prompt, so that you know what to enter. Combine this with a *DFN Fx* and you will end up with a program that is a lot easier to use, because you can now restart it many times without going through the above *RUN <MEM> F5..F5 <FIL>* sequence over and over again.

An example of such an input routine is:

```
0000 CE 'X! (All digits)'     Information about the function of the program
0016 DFN CLR                  Clear the Fx keys
0018 DFN F1: X @01 RTN        The user defined key to enter your input
0026 LBL 01                   Start of the program
0029 STO xxx
0032 Rest of program
```

Programs requiring up to about ten [unrelated] input values

If your program needs up to about ten input values, you can use a variation on the single input DFN Fx routine, you define an Fx key for each of the variables that you need and one more to start the program. This approach works out fine if your program doesn't need more than four input values, but what do you do if you need to enter more than four variables? The solution to this problem is simple, you define F5 as —> and use it to jump to a next data entry menu.

If you are using this method, the program should only store your input. The reasons for not performing any calculations on it, except for direct functions such as SIN, LOG, X^2 etc, are

- they tend to slow down the rate of data entry
- calculations may affect values you already entered, so that you have to start all over again, which can be quite frustrating if your last (and incorrect) entry messes up 10 other 13 digit numbers ...

The advantage of the one Fx key per variable is clarity, the big disadvantage is the fact that it consumes lots of memory - entering 11 values using this approach takes 183 bytes, 15 * 7 for the 15 DFN Fx'es, 11 * 6 for the routines that store the input and 3 * 4 for the LBLs and RTNs around the DFNs. This large amount of memory may not be very nice, but for some problems - quadratic, cubic and quartic equations, conic sections [= $Ax^2 + Bxy + Cy^2 + Dx + Ey + F$] and the like - there really isn't a much shorter alternative - unless you start messing around in the systemregisters.

Data Entry - (cont)

An example of an input routine for a 7 variable problem is:

```
0000 LBL M1              0063 DFN F4:EOD@GO
0003 DFN F1: A @01       0070 DFN F5:-->@M1
0010 DFN F2: B @02       0077 RTN
0017 DFN F3: C @03       0078 LBL 01 STO A RTN
0024 DFN F4: D @04       0084 LBL 02 x^2 STO B RTN
0031 DFN F5:-->@M2       0090 LBL 03 STO C RTN
0038 RTN                 0096 LBL 04 STO D RTN
0039 LBL M2              0102 LBL 05 SIN STO E RTN
0042 DFN F1: E @05       0108 LBL 06 STO F RTN
0049 DFN F2: F @06       0114 LBL 07 LOG STO G RTN
0056 DFN F3: G @07       0121 LBL GO ....
```

You will see that I added a --> to the second menu. It allows you to enter your data in full random order. The EOD above F4 on the second menu is the term TI uses in its application ROMs, it stands for End Of Data. You can of course use something more meaningful.

Programs requiring large amounts of related data

If a program needs lot of data, the data will usually be related in one way or another. Examples of related data are

- elements of a vector or matrix
- coefficients of a polynomial
- cash flows for a number of periods

In these cases it would be foulish to spend a lot of programsteps to enter your data. Using the approach described above, you would need about ***300(!)*** programsteps to enter the variables in a 4 x 4 equation system...

To prevent such a waste of memory, we should look for another method to enter our data. Of course such a method exists, but it is a little more complex than the two previous methods, because we have to use indirect addressing to store the data. [The basics of indirect addressing are described in Chapter 6 of the ***TI-95 Programming Guide***]

The key ingredients for a data entry routine that uses indirect adressing are,

- a register containing a pointer to the registers you use to store your data
- a counter register to ensure that you don't enter to many values
- the alpha register, to provide a prompt so thay you know what to enter

Besides these three necessary routines, you may sometimes want to add a little more code

- to print the input with a suitable descriptor
- to edit the input, so that you can make corrections

A typical routine containing all five ingredients may look like the one on the next page. I will assume that the program needs N inputvalues, that have to be stored from R 023. The names of the variables are X[1] .. X[N]. Because the routine is written with labels, it can be put at any place in memory.

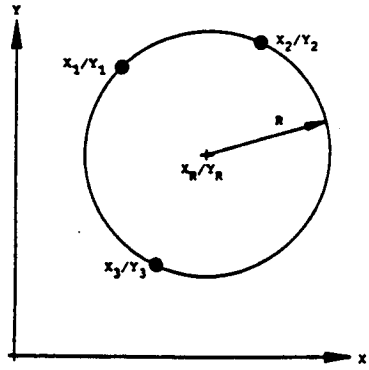
The example clearly shows the advantage of indirect addressing, a very short data entry ***AND*** editing routine, that can accept a large amount of input. It can also be tailored to specific applications by changing the steps that are involved in the starting register and the prompt.

Data Entry - (cont)

```
0000 CMS                                  Clear all memories
0001 DFN CLR                              Clear the Fx keys
0003 DFN F1: N @01                        Define F1 to enter N
0010 RTN
0011 LBL 01
0014 STO A                                Store N
0016 23 STO B                             First register to be used for storing the X[n]
0020  1 STO C                             We want to start with X[1]
0023 RCL A STO D                          Initialisation of a counter, so that we can enter only N X[n]'s
0027 `N =` COL 16 MRG A PRT               Print N
0035 DFN F1:ENT@03                        Redefine F1 as ENT, to enter the X[n]
0042 LBL 02
0045 CE `X[` COL 05 MRG C COL 05 `]` RTN  Display the prompt for the X[n]
0056 LBL 03
0059 STO IND B                            Store X[n] in the register pointed to by B
0062 OLD `=` COL 16 MRG = PRT             Get the prompt back, add a `=`, X[n] and print it
0069 INC B INC C                          Increment the storage pointer and the prompt counter
0073 DSZ A GTL 02                         Repeat N times
0078 CE `EDIT?`                           Ask if the user wants to edit any values
0084 Y/N GTL 04 GTL 07                    If YES GTL 04, if NO start the actual program
0091 LBL 04
0094 DFN F1: i @05                        Define F1 to point the edit routine to X[i]
0101 DFN F2:ENT@06                        Define F2 to enter the correct value for X[i]
0108 DFN F3:CAL@07                        Define F3 to start the actual program
0115 RTN
0116 LBL 05
0119 STO B STO C                          Store i, both for the prompt and as a pointer
0123 22 ST+ B                             Adjust the pointer, as we started with X[1] in R 023
0127 RCL IND B                            Recall X[i] and display it with a nice description. Recall is useful
0130 SBL 02 `=` COL 16 MRG = RTN          because it allows you to press <ENT> if you find that it is OK.
0139 LBL 06
0142 STO IND B                            Store the new value
0145 SBL 02 `=` COL 16 MRG = PRT RTN      Print it with a description
0155 LBL 07
0158 DFN CLR                              Clear the data entry and/or edit Fx keys
0160 Actual program ...
```

CIRCLE THROUGH THREE POINTS - P. Hanson.

When work with the programming challenge on V12N4P26 required a routine for finding the circle which passes through three points I looked through past issues of our newsletter for an existing program. I found one for the TI-57 on V7N1/2P27. It was written by Ingvar Magnusson, and was originally published in the Swedish newsletter Programbiten. The program listing appears at the right. To run:



1. Enter X1 and press STO 0.
2. Enter Y1 and press STO 1.
3. Enter X2 and press STO 2.
4. Enter Y2 and press STO 3.
5. Enter X3 and press STO 4.
6. Enter Y3 and press STO 5.
7. Press RST R/S and see R in the display.
8. Press RCL 4 and see XR in the display.
9. Press RCL 5 and see YR in the display.

```
00  33  4  RCL 4          25  23    X²
01 -34  0  INV SUM 0      26  75    -
02 -34  2  INV SUM 2      27  33  0 RCL 0
03  33  5  RCL 5          28  39  7 PRD 7
04 -34  1  INV SUM 1      29  23    X²
05 -34  3  INV SUM 3      30  85    =
06  33  2  RCL 2          31  39  3 PRD 3
07  55     ×              32  39  2 PRD 2
08  33  1  RCL 1          33  33  3 RCL 3
09  65     -              34 -34  1 INV SUM 1
10  33  0  RCL 0          35  33  7 RCL 7
11  55     ×              36 -34  2 INV SUM 2
12  33  3  RCL 3          37  33  6 RCL 6
13  85     =              38 -39  1 INV PRD 1
14  32  6  STO 6          39 -39  2 INV PRD 2
15  34  6  SUM 6          40  33  1 RCL 1
16  33  2  RCL 2          41  34  4 SUM 4
17  23     X²             42  23    X²
18  75     -              43  75    -
19  33  3  RCL 3          44  33  2 RCL 2
20  23     X²             45  34  5 SUM 5
21  85     =              46  23    X²
22  32  7  STO 7          47  85    =
23  38  1  EXC 1          48  24    ΓX
24  39  1  PRD 1          49  81    R/S
```

Conversion for the TI-59 is straightforward since there are corresponding TI-59 commands for each TI-57 command. The listing which follows provides for data entry with the User Defined keys and printout with the PC-100. A printout for a sample problem appears at the right. To use the program:

1. Enter X1 and press x t. Enter Y1 and press A.
2. Enter X2 and press x t. Enter Y2 and press B.
3. Enter X3 and press x t. Enter Y3 and press C.
4. To solve press D.

```
        -6.
         2.

         4.
         4.

         0.
        -6.

-.3913043478          H
-.0434782609          K
 5.969360964          R
```

```
000  76 LBL    027  32 X:T    054  03  03    081  48 EXC    108  06  06    135  69 DP
001  11  A     028  99 PRT    055  43 RCL    082  01  01    109  22 INV    136  06  06
002  98 ADV    029  42 STO    056  02  02    083  49 PRD    110  49 PRD    137  02  2
003  32 X:T    030  04  04    057  65  ×     084  01  01    111  01  01    138  06  6
004  99 PRT    031  32 X:T    058  43 RCL    085  33 X²     112  22 INV    139  69 DP
005  42 STO    032  99 PRT    059  01  01    086  85  +     113  49 PRD    140  04  04
006  00  00    033  42 STO    060  75  -     087  43 RCL    114  02  02    141  43 RCL
007  32 X:T    034  05  05    061  43 RCL    088  00  00    115  43 RCL    142  05  05
008  99 PRT    035  91 R/S    062  00  00    089  49 PRD    116  01  01    143  69 DP
009  42 STO    036  76 LBL    063  65  ×     090  07  07    117  44 SUM    144  06  06
010  01  01    037  14  D     064  43 RCL    091  33 X²     118  04  04    145  03  3
011  91 R/S    038  98 ADV    065  03  03    092  95  =     119  33 X²     146  05  5
012  76 LBL    039  43 RCL    066  95  =     093  49 PRD    120  85  +     147  69 DP
013  12  B     040  04  04    067  42 STO    094  03  03    121  43 RCL    148  04  04
014  98 ADV    041  22 INV    068  06  06    095  49 PRD    122  02  02    149  32 X:T
015  32 X:T    042  44 SUM    069  44 SUM    096  02  02    123  44 SUM    150  69 DP
016  99 PRT    043  00  00    070  06  06    097  43 RCL    124  05  05    151  06  06
017  42 STO    044  22 INV    071  43 RCL    098  03  03    125  33 X²     152  98 ADV
018  02  02    045  44 SUM    072  02  02    099  22 INV    126  95  =     153  91 R/S
019  32 X:T    046  02  02    073  33 X²     100  44 SUM    127  34 ΓX     154  00  0
020  99 PRT    047  43 RCL    074  85  +     101  01  01    128  32 X:T    155  00  0
021  42 STO    048  05  05    075  43 RCL    102  43 RCL    129  02  2     156  00  0
022  03  03    049  22 INV    076  03  03    103  07  07    130  03  3     157  00  0
023  91 R/S    050  44 SUM    077  33 X²     104  22 INV    131  69 DP    158  00  0
024  76 LBL    051  01  01    078  95  =     105  44 SUM    132  04  04    159  00  0
025  13  C     052  22 INV    079  42 STO    106  02  02    133  43 RCL
026  98 ADV    053  44 SUM    080  07  07    107  43 RCL    134  04  04
```

Circle through three points - (cont)

When operating without a PC-100 the calculator will stop with the radius of the circle in the display. Then press RCL 04 to see the x coordinate of the center of the circle (H) in the display, and press RCL 05 to see the y coordinate of the center of the circle (K).

```
R =  5.969360964
H = -.3913043478
K = -.0434782609
```

Conversion from the TI-59 to the TI-95 is also straightforward. The only changes involve replacement of the User Defined Key control of the TI-59 with Function Key control in response to prompts in the windows of the TI-95. The program listing appears below. A sample printout appears at the right. To use the program, enter the coordinates of the three points using the function keys. Press F5 to see the prompts for the third point. Press F3 (ANS) to obtain the solution. Without a printer use the prompts to call the solution parameters (H, K, and R) to the display.

```
0000 CLR
0001 LBL XX
0004 DFN F1: X1@X1
0011 DFN F2: Y1@Y1
0018 DFN F3: X2@X2
0025 DFN F4: Y2@Y2
0032 DFN F5:-->@YY HLT
0040 LBL YY
0043 DFN F1: X3@X3
0050 DFN F2: Y3@Y3
0057 DFN F3:ANS@AN
0064 DFN F5:-->@XX HLT
0072 LBL X1 STO 000 HLT
0079 LBL Y1 STO 001 HLT
0086 LBL X2 STO 002 HLT
0093 LBL Y2 STO 003 HLT
0100 LBL X3 STO 004 HLT
0107 LBL Y3 STO 005 HLT
0114 LBL AN ADV DFN CLR
0120 DFN F1: H @HH
0127 DFN F2: K @KK
0134 DFN F3: R @RR
0141 RCL 004 ST- 000
0147 ST- 002 RCL 005
0153 ST- 001 ST- 003
0159 RCL 002 * RCL 001 -
0167 RCL 000 * RCL 003 =
0175 STO 006 ST+ 006
0181 RCL 002 x^2 +
0186 RCL 003 x^2 =
0191 STO 007 EXC 001
0197 ST* 001 x^2 +
0202 RCL 000 ST* 007 x^2
0209 = ST* 003 ST* 002
0216 RCL 003 ST- 001
0222 RCL 007 ST- 002
0228 RCL 006 ST/ 001
0234 ST/ 002 RCL 001
0240 ST+ 004 x^2 +
0245 RCL 002 ST+ 005 x^2
0252 = SQR STO 008
0257 LBL RR CLR RCL 008
0264 `R = ` COL 16 MRG =
0272 PRT HLT
0274 LBL HH CLR RCL 004
0281 `H = ` COL 16 MRG =
0289 PRT HLT
0291 LBL KK CLR RCL 005
0298 `K = ` COL 16 MRG =
0306 PRT HLT
```

For a conversion for the TI-74 the data input and data output routines must be changed to accommodate the TI-74 and the associated peripherals. For example, in the listing on the next page lines 10 through 70 are similar to the output selection routines used in TI-74 programs in earlier issues. A "brute force" translation of the solution sequence can be obtained by designating variables A through H to correspond to data registers R00 through R07 of the TI-59. It will also be helpful to designate a variable X to correspond to the display register of the TI-59. Some example translations follow:

a. Steps 055-070 of the TI-59 program which calculate two products (R02 x R01 and R00 x R03) and place twice the difference in R06 can be translated into the statements on lines 220 and 225 of the BASIC program for the TI-74.

b. Translation of TI-59 commands such as the EXC 01 at steps 081-082, which exchange the contents of the display register with data register R01, cannot be directly translated since there is no equivalent command in BASIC. One alternative is to use yet another variable in the BASIC program, say register Y, and write a sequence of statements Y = X, X = B and B = Y. In this translation I substituted equivalent commands based on more thorough analysis of what was actually being done in the program.

Circle through three points - (cont)

The reader is invited to compare other segments of the BASIC program below with the TI-59 program listing on page 24 and satisfy himself of the equivalence of the two programs.

```
 10 A$="Circle Thru 3 Poi
 nts":PRINT A$:PAUSE 1
 20 INPUT "Use Printer? Y
 /N ";Z$
 30 IF Z$="Y"OR Z$="y"THE
 N PN=1 ELSE 100
 35 PRINT "Device Numbers
 :":PAUSE 1
 40 PRINT "For the HX-100
 0 enter 10":PAUSE 1
 45 PRINT "For the PC-324
  enter 12":PAUSE 1
 50 INPUT "Enter device n
 umber ";D$
 55 OPEN #1,D$,OUTPUT
 60 IF D$="10"THEN PRINT
 #1,CHR$(18)
 65 PRINT #1:PRINT #1,A$
 70 PRINT #1
 100 X$="X(1) = ":INPUT X
 $;A:PRINT #PN,X$;A
 105 X$="Y(1) = ":INPUT X
 $;B:PRINT #PN,X$;B
 110 PRINT #PN
 115 X$="X(2) = ":INPUT X
 $;C:PRINT #PN,X$;C
 120 X$="Y(2) = ":INPUT X
 $;D:PRINT #PN,X$;D
 125 PRINT #PN
 130 X$="X(3) = ":INPUT X
 $;E:PRINT #PN,X$;E
 135 X$="Y(3) = ":INPUT X
 $;F:PRINT #PN,X$;F
 140 PRINT #PN
 200 A=A-E
 205 C=C-E
 210 B=B-F
 215 D=D-F
 220 G=C*B-A*D
 225 G=G+G
 230 H=C*C+D*D
 235 X=B:B=H
 240 B=X*B
 245 H=A*H
 250 X=X*X+A*A
 255 D=X*D
 260 C=X*C
 265 B=B-D
 270 C=C-H
 275 B=B/G
 280 C=C/G
 285 E=E+B
 290 F=F+C
 295 R=SQR(B*B+C*C)
 400 PAUSE ALL
 410 PRINT #PN," h = ";E
 420 PRINT #PN," k = ";F
 430 PRINT #PN," r = ";R
 440 PAUSE 0:PRINT #PN
 999 END
```

Lines 200-295 will be used in the solution of the programming challenge from V12N4P26.

BEST FIT CIRCLE - At one time it appeared that we might need a routine for finding the best fit circle through four or more points in the solution to the programming challenge on V12N4P26. Appropriate equations are defined in William Kolb's Curve Fitting for Programmable Calculators. The "Circle Best Fit" program in Maurice Swinnen's Statistical Library book (see V12N3P4) provides an equivalent program in Sharp BASIC which is readily translatable for TI-74 BASIC. The resulting program listing is on page 27. A complete set of prompts are provided. Sample problems are illustrated in the right hand column on page 27.

Lines 10 through 70 provide for selection of the display, the PC-324 or the HX-1000 for output.

Lines 100 through 190 provide for data entry using the string input techniques used in Maurice Swinnen's books.

Lines 200 through 370 are a direct translation from Maurice's book.

Lines 400 through 440 provide for output of the solution to the display or to the printer based on the value of the variable PN which was defined in line 30. The PAUSE ALL statement in line 400 combined with a value of PN of zero causes calculation to stop as each parameter is displayed when a printer is not used.

Lines 500 through 690 provide an ability to calculate points on the defined circle.

Lines 700-770 provide for adding points to the input array for a revised solution.

Best Fit Circle - (cont)

Program Listing

```
10 A$="Best Fit Circle":
PRINT A$:PAUSE 1
20 DIM X(50),Y(50)
25 INPUT "Use Printer? Y
/N ";Z$
30 IF Z$="Y"OR Z$="y"THE
N PN=1 ELSE 100
35 PRINT "Device Numbers
:":PAUSE 1
40 PRINT "For the HX-100
0 enter 10":PAUSE 1
45 PRINT "For the PC-324
 enter 12":PAUSE 1
50 INPUT "Enter device n
umber ";D$
55 OPEN #1,D$,OUTPUT
60 IF D$="10"THEN PRINT
#1,CHR$(18)
65 PRINT #1:PRINT #1,A$
70 PRINT #1
100 PRINT "End Input by
Entering E":PAUSE 1
110 N=1
120 X$="X = "
125 Y$="Y = "
130 INPUT X$;XX$:IF XX$=
"E"OR XX$="e"THEN 190
135 INPUT Y$;YY$:IF YY$=
"E"OR YY$="e"THEN 190
140 X(N)=VAL(XX$)
145 Y(N)=VAL(YY$)
150 IF PN=0 THEN 180
155 PRINT #PN,X$;X(N)
160 PRINT #PN,Y$;Y(N)
170 PRINT #PN
180 N=N+1:GOTO 130
190 N=N-1
200 S6=0:S7=0:S8=0:S9=0:
T0=0:T1=0:U6=0:U9=0:V0=0
:V2=0
210 FOR I=1 TO N
220 U=X(I):V=Y(I)
230 S6=S6+U:S7=S7+U*U:S8
=S8+V:S9=S9+V*V:T0=T0+U*
V:T1=T1+1
240 U6=U6+V*U*U:U9=U9+U*
V*V:V0=V0+U*U*U:V2=V2+V*
V*V
250 NEXT I
300 R5=T0*T1-S6*S8:R6=(S
7+S9)*S8-(U6+V2)*T1
310 R7=S8*S8-S9*T1:R8=(S
7+S9)*S6-(U9+V0)*T1:R9=S
6*S6-S7*T1
320 IF R8=0 THEN PRINT "
Solution Undefined":STOP
330 S0=R7*R9-R5*R5
340 H=(R5*R6+R7*R8)/(2*S
0)
350 K=(R5*R8+R6*R9)/(2*S
0)
360 AA=2*(H*S6+K*S8):BB=
(H*H+K*K)*T1
370 R=SQR((S7+S9-AA+BB)/
T1)
400 PAUSE ALL
410 PRINT #PN," h = ";H
420 PRINT #PN," k = ";K
430 PRINT #PN," r = ";R
440 PAUSE 0:PRINT #PN
500 P$="Predict y for x"
505 PRINT #PN,P$:PRINT #
PN
510 INPUT P$&" = ";XX$
520 IF XX$="E"OR XX$="e"
THEN 600
530 XX=VAL(XX$)
540 YY=SQR(ABS(R*R-(XX-H
)^2))+K
550 PAUSE ALL
560 PRINT #PN,X$;XX
570 PRINT #PN,Y$;YY
580 PAUSE 0:PRINT #PN
590 GOTO 510
600 P$="Predict x for y"
605 PRINT #PN,P$:PRINT #
PN
610 INPUT P$&" = ";YY$
620 IF YY$="E"OR YY$="e"
THEN 700
630 YY=VAL(YY$)
640 XX=SQR(ABS(R*R-(YY-K
)^2))+H
650 PAUSE ALL
660 PRINT #PN,Y$;YY
670 PRINT #PN,X$;XX
680 PAUSE 0:PRINT #PN
690 GOTO 610
700 P$="Add more data (Y
/N) ? "
710 INPUT P$;Z$
720 IF Z$="N"OR Z$="n"TH
EN 999
730 PRINT #PN,"Additiona
l Input Data"
740 PRINT #PN
750 N=N+1
760 GOTO 120
770 GOTO 120
999 END
```

Examples

```
Best Fit Circle

X =  12
Y =  1

X =  14
Y =  3

X =  15
Y =  6

X =  14
Y =  9

 h =  9.357142857
 k =  5.948979592
 r =  5.577611132

Predict y for x

X =  4
Y =  7.501643866

X =  5
Y =  9.431084137

Predict x for y

Y =  9
X =  14.02629914

Additional Input Data

X =  12
Y =  11

X =  9
Y =  12

 h =  8.976331361
 k =  6.035502959
 r =  5.89669835

Predict y for x

X =  4
Y =  9.198915297

Predict x for y

Y =  9
X =  14.07366485
```

SMALLEST CIRCLE TO ENCLOSE FOUR OR MORE POINTS - This programming challenge was proposed by Don Laughery in V12N4P26. He stated that a solution would have application in the field of positional tolerancing. Larry Leeds and Peter Messer responded with assistance from the editor.

The Editor's Note which accompanied the challenge stated that the challenge was clearly for more than three points since algebraic solutions for a circle through three points were well known. Our first insight was recognition that in many cases the circle through three points may not be the smallest circle to enclose the three points. Consider the case of three points where the third point is inside the circle with a diameter equal to the distance between the other two, and with a center at the mid-point between the two points. This is then the minimum diameter circle for the three points.

It was evident that the first phase in any solution would involve finding the pair of points which were furthest from each other (call them P1 and P2), and testing the remaining points (call them P3, P4, etc.) to determine if they were inside the minimum radius circle through P1 and P2. If all other points were inside the defined circle the problem was solved.

Larry and Peter then proposed algorithms for solution when any of the remaining points (P3, P4, etc.) were outside the minimum circle defined by P1 and P2. The proposed algorithms involved testing the distance of the remaining points from either P1 or P2, or from the center of the minimum circle through P1 and P2, and then solving for the circle through P1, P2 and a selected third point. A little reflection will reveal that the circles so defined will have their centers on the perpendicular bisector of the line segment between P1 and P2.

The editor programmed the algorithms for the TI-74. The solution was ready for publication when it was found that there were some cases where a smaller diameter circle could be found to enclose all of the points by using only one of the most distant points (P1 or P2) and two other points. One example is the set of points (4,4), (0,-6), (-5,2), and (5,3). In this case the most distant points are (4,4) and (0,-6) but the circle through the last three points yields a smaller diameter which will enclose the four points.

At this point the editor wrote a new program for the TI-74 which finds the minimum circle defined by P1 and P2, and also finds the parameters (center and radius) for each circle defined by any combination of three points in the ensemble. The program also examines each circle so defined to determine whether or not the remaining points are within the circle. If any point is not, the radius of the circle is increased until all points are included. Finally, the circle with minimum radius is selected as the solution. The program on page 29 mechanizes that algorithm. An example solution is also provided on page 29, where it is yet to be proved that even this solution is the one which yields the minimum radius which will include all of the points.

Lines 10 through 70 provide for selection of the display, the PC-324 or the HX-1000 for output.

Lines 100 through 190 provide for data entry using a modified form of the string input techniques used in Maurice Swinnen's books. In this case the end of input data is signalled by pressing ENTER with a null string (no input) in the display, rather than by entering an "E" or an "e" and pressing ENTER. User's will find this is more convenient.

Lines 200 through 245 identify the pair of most distant points (P1 and P2) and find the center of the minimum circle through the two points.

Smallest Circle - (cont)

Lines 300 through 335 examine each of the remaining points (P3, P4, etc.). If a point is outside the minimum circle the radius is increased to include that point. The center of the circle is not changed. If none of the remaining points is outside the minimum circle through P1 and P2 the program proceeds to print the solution.

Lines 345 through 760 examines each possible set of three points from the ensemble in order, find the parameters which define the circle through the three points, examine the remaining points to determine whether they are inside the defined circle or not and increases the radius (actually the square of the radius) to include all points. If the resulting circle is smaller than the previously selected circle then its parameters are used as the basis for examining subsequent candidates.

Lines 400 through 460 use the solution for a circle through three points from lines 200 through 295 of the program on page 26.

Lines 800 through 850 print the solution, or display the solution if a printer is not used.

```
10 A$="Smallest Circle":
PRINT A$:PAUSE 1
20 DIM X(10),Y(10)
25 INPUT "Use Printer? Y
/N ";Z$
30 IF Z$="Y"OR Z$="y"THE
N PN=1 ELSE 100
35 PRINT "Device Numbers
:":PAUSE 1
40 PRINT "For the HX-100
0 enter 10":PAUSE 1
45 PRINT "For the PC-324
 enter 12":PAUSE 1
50 INPUT "Enter device n
umber ";D$
55 OPEN #1,D$,OUTPUT
60 IF D$="10"THEN PRINT
#1,CHR$(18)
65 PRINT #1:PRINT #1,A$
70 PRINT #1
100 PRINT "End Input by
Entering "&CHR$(255):PAU
SE 2
110 N=1
120 X$="X = "
125 Y$="Y = "
130 INPUT X$;XX$:IF XX$=
""THEN 190
135 INPUT Y$;YY$:IF YY$=
""THEN 190
140 X(N)=VAL(XX$)
145 Y(N)=VAL(YY$)
150 IF PN=0 THEN 180
155 PRINT #PN,X$;X(N)
160 PRINT #PN,Y$;Y(N)
170 PRINT #PN
180 N=N+1:GOTO 130
190 N=N-1
195 PRINT "Solving"
200 M1=0
205 FOR I=1 TO N
210 FOR J=2 TO N-1
215 D2=(X(I)-X(J))^2+(Y(
I)-Y(J))^2
220 IF D2>M1 THEN M1=D2:
S=I:T=J
225 NEXT J
230 NEXT I
235 HH=(X(S)+X(T))/2
240 KK=(Y(S)+Y(T))/2
245 M2=M1/4
300 Z=0
305 FOR I=1 TO N
310 IF I=S OR I=T THEN 3
25
315 D2=(X(I)-HH)^2+(Y(I)
-KK)^2
320 IF D2>M2 THEN M2=D2:
Z=1
325 NEXT I
335 IF Z=0 THEN 800
345 FOR I=N TO 3 STEP -1
350 FOR J=(I-1)TO 2 STEP
 -1
355 FOR L=(J-1)TO 1 STEP
 -1
365 A=X(I):B=Y(I)
370 C=X(J):D=Y(J)
375 E=X(L):F=Y(L)
400 A=A-E:C=C-E
405 B=B-F:D=D-F
410 G=C*B-A*D:G=G+G
415 H1=C*C+D*D
420 X1=B:B=H1
425 B=X1*B:H1=A*H1
430 X1=X1*X1+A*A
435 D=X1*D:C=X1*C
440 B=B-D:C=C-H1
445 B=B/G:C=C/G
450 E=E+B:F=F+C
455 R2=B*B+C*C
460 H=E:K=F
705 FOR M=1 TO N
710 IF M=L OR M=J OR M=I
 THEN 725
715 D2=(X(M)-H)^2+(Y(M)-
K)^2
720 IF D2>R2 THEN R2=D2
725 NEXT M
730 IF R2<M2 THEN M2=R2:
HH=H:KK=K
735 NEXT L
755 NEXT J
760 NEXT I
800 PAUSE ALL
810 PRINT #PN," h = ";HH
820 PRINT #PN," k = ";KK
830 PRINT #PN," r = ";SQ
R(M2)
840 IF PN=1 THEN PRINT #
1
850 PAUSE 0
999 END
```

```
Smallest Circle

X = -7
Y =  0

X =  0
Y =  6

X =  3.5
Y =  5

X =  0
Y = -6

X = -2
Y =  5

 h = -.6913265306
 k =  .2767857143
 r =  6.314742376
```

BIORHYTHMS ON THE TI-74 - A biorhythms program for the TI-95 was published in the Volume 2, Number 2 issue of Programmable Calculator News. Page 2 of this issue notes that there were errors in the listing at lines 210 and 338. When members asked about a similar program for the TI-74 I decided that the conversion would provide another good example of translation from the TI-59 or TI-95 to the TI-74. The methodology I selected was the same as for the translation of the program for a circle through three points on pages 25 and 26. The task was easier in this case because the TI-95 program was already using alphabetic addressing for the data registers and the TI-74 translation simply assigned a letter variable to match each TI-95 data register. The variable X was assigned to simulate the display register of the TI-95. A listing of the translation for the TI-74 and some sample printouts follow.

Program Listing

```
1000 A$="Biorhythms":PRI
NT A$:PAUSE 1
1010 INPUT "Use Printer?
 Y/N ";Z$
1020 IF Z$="Y"OR Z$="y"T
HEN PN=1 ELSE 1100
1030 PRINT "Device Numbe
rs:":PAUSE 1
1040 PRINT "For the HX-1
000 enter 10":PAUSE 1
1050 PRINT "For the PC-3
24 enter 12":PAUSE 1
1060 INPUT "Enter device
 number ";D$
1070 OPEN #1,D$,OUTPUT
1080 IF D$="10"THEN PRIN
T #1,CHR$(18)
1090 PRINT #1:PRINT #1,A
$:PRINT #1
1100 B$="Birthday ":T$="
Today ":I$="(mmdd.yyyy)?
 ":IF PN=0 THEN PAUSE AL
L
1105 DEG:Z=1:INPUT B$&I$
;D$
1110 GOTO 1800
1140 PRINT #PN,B$&"
 ";D$:IF PN=1 THEN PRINT
 #1
1145 E=X
1150 Z=2:INPUT T$&I$;D$
1155 GOTO 1800
1190 PRINT #PN,T$&"
    ";D$:IF PN=1 THEN PR
INT #1
1195 F=X
1200 G=F-E
1210 PRINT #PN,G;"days o
ld"
1220 IF PN=1 THEN PRINT
#1
1230 D$=" Days into cycl
e "
1240 P$=" % through cycl
e "
1250 M$=" Amplitude
  "
1300 PRINT #PN,"Physical
 Cycle:"
1305 X=G/23:X=X-INT(X):M
=X:X=23*X
1310 X=X+.1:X=INT(X)
1315 H=X:X=X/23:X=1000*X
1320 X=INT(X):X=X/10:T=X
1325 GOSUB 1700
1400 PRINT #PN,"Emotiona
l Cycle:"
1405 X=G/28:X=X-INT(X):M
=X:X=28*X
1410 X=X+.1:X=INT(X)
1415 H=X:X=X/28:X=1000*X
1420 X=INT(X):X=X/10:T=X
1425 GOSUB 1700
1500 PRINT #PN,"Intellec
tual Cycle:"
1505 X=G/33:X=X-INT(X):M
=X:X=33*X
1510 X=X+.1:X=INT(X)
1515 H=X:X=X/33:X=1000*X
1520 X=INT(X):X=X/10:T=X
1525 GOSUB 1700
1600 INPUT "Another solu
tion (Y/N)? ";Z$
1605 IF Z$="Y"OR Z$="y"T
HEN 1610 ELSE END
1610 IF PN=1 THEN 1090 E
LSE 1100
1700 IF PN=1 THEN PRINT
#1
1705 PRINT #PN,D$;H
1710 IF PN=1 THEN PRINT
#1
1715 PRINT #PN,P$;T
1720 IF PN=1 THEN PRINT
#1
1725 M=INT(1000*SIN(360*
M))/1000
1730 PRINT #PN,M$;M
1735 IF PN=1 THEN PRINT
#1
1740 RETURN
1800 X=VAL(D$):B=X:GOSUB
 1950:X=0
1805 GOSUB 1950:IF X<A T
HEN 1900
1810 X=X-INT(X):B=B-X:X=
10000*X
1815 D=X:GOSUB 1950:X=15
81:IF X>=A THEN 1900
1820 X=32:GOSUB 1950:X=B
:X=X/100:B=X
1825 X=X-INT(X):B=B-X:X=
100*X:C=X:IF X>=A THEN 1
900
1830 X=13:GOSUB 1950:X=B
:IF X>=A THEN 1900
1835 X1=365*D+C+31*B-31
1840 X=3:GOSUB 1950:X=B:
IF X>=A THEN 1960
1845 D=D-1
1850 X=X1+INT(D/4)
1855 X1=INT(D/100)
1860 X1=.75+.75*X1
1865 X1=INT(X1)
1870 X=X-X1
1895 ON Z GOTO 1140,1190
1900 E$="Entry error: tr
y again (Y/N)?"
1910 INPUT E$;E$
1920 IF E$="N"OR E$="n"T
HEN 1999
1930 ON Z GOTO 1100,1150
1950 T=X:X=A:A=T:RETURN
1960 X1=X1-INT(.4*B+2.3)
:GOTO 1850
1999 END
```

Biorhythms for the TI-74 - (cont)

Sample Printouts

Biorhythms	
Birthday	912.1951
Today	608.1988
13419 days old	
Physical Cycle:	
Days into cycle	10
% through cycle	43.4
Amplitude	.398
Emotional Cycle:	
Days into cycle	7
% through cycle	25
Amplitude	.999
Intellectual Cycle:	
Days into cycle	21
% through cycle	63.6
Amplitude	-.756

Biorhythms	
Birthday	913.1946
Today	423.1949
953 days old	
Physical Cycle:	
Days into cycle	10
% through cycle	43.4
Amplitude	.398
Emotional Cycle:	
Days into cycle	1
% through cycle	3.5
Amplitude	.222
Intellectual Cycle:	
Days into cycle	29
% through cycle	87.8
Amplitude	-.691

Biorhythms	
Birthday	222.1954
Today	901.1977
8592 days old	
Physical Cycle:	
Days into cycle	13
% through cycle	56.5
Amplitude	-.399
Emotional Cycle:	
Days into cycle	24
% through cycle	85.7
Amplitude	-.782
Intellectual Cycle:	
Days into cycle	12
% through cycle	36.3
Amplitude	.755

Some comments on the translation:

I have not provided a copy of the TI-95 program. If you have not yet subscribed to Programmable Calculator News you should do so as soon as possible.

a. Lines 1000 through 1090 provide for selection of the display, the PC-324 or the HX-1000 for output.

b. Line 1100 defines some string variables which are used more than once in the program.

c. Lines 1105-1195 are the equivalent of the LBL AA portion of the TI-95 program (steps 0270 ff).

d. Lines 1800-1895 are the equivalent of the LBL E' portion of the TI-95 program. In this routine I tried to make a true "brute force" translation. For example, with a value in the display register of the TI-95 a STO B command is translated into B = X for the TI-74. Similarly, a FRC command in the TI-95 is translated into X = X - INT(X) for the TI-74. Where nesting of parenthesis was involved it was necessary to define additional variables, e.g., X1, to hold intermediate values as in line 1835 of the TI-74 program. This is equivalent to the storage of nested values in the hierarchy registers of the TI-59 or TI-95.

Biorhythms for the TI-74 - (cont)

e. Lines 1900-1930 are the equivalent of the LBL M2 portion of the TI-95 program and provide for input error recovery.

f. Line 1950 provides the equivalent of the EXC A command which is used many times in the TI-95 program.

g. Line 1960 is the equivalent of the LBL M1 portion of the TI-95 program. If you are going to pursue this brute force method of conversion you should be sure you understand the conditions at the call and return for this subroutine.

h. Lines 1700-1740 are the equivalent of the LBL BB portion of the TI-95 program. Lines 1725-1730 are not in the TI-95 program. They were added to provide the "Amplitude" output for each cycle. I have found that parameter to be of interest to people to study biorhythms. The mechanization is equivalent to that in the Biorhythms program (LE-21) in the Leisure Library module for the TI-59.

i. Lines 1600-1610 provide the capability to obtain a second solution without going back through the output device selection routine.

MORE ON TRANSLATIONS FROM THE TI-59 TO THE TI-95 - P. Hanson.

HIR and the Hierarchy Registers

Several members have asked for explanation of the hierarchy registers and HIR commands of the TI-59. The hierarchy registers were normally used to store nested arithmetic operations, to store results from operations such as P/R, D.MS, and the statistics functions, and to store printer output as defined by the OP 01 through OP 04 commands. The HIR commands which provide direct access to the hierarchy registers were not mentioned in the Personal Programming manual. One of the earliest descriptions of the commands appeared in the September 1977 issue of 52 Notes, where the "discovery" was credited to Heinrich Schnepf. Some very complex operations can be attained through use of the HIR commands--one example occurs in the calendar printing programs. More often programmers use the commands to expand the number of data registers. An example appears in the extended calculator diagnostic program on V12N1P5 where the 1 HIR 37 sequence at steps 143-145 increments hierarchy register 7 at the completion of each phase of the test, and the HIR 17 sequence at steps 147-148 retrieves the incremented value to set up the test code for the next phase. The major portion of HIR useage in programs can be described by noting that in an HIR XY command the Y defines the hierarchy register used (1 through 8) and the X defines the operation according to the following code:

X	OPERATION
0	STO
1	RCL
3	SUM
4	Prd
5	INV SUM
6-9	INV Prd

Translation from the TI-59 to the TI-95 - (cont)

The missing value of 2 for X is not an oversight. An HIR 20 code appears in the firmware for the TI-59, and seems to be a specialized branching command. There has been no report of successful use of an HIR 2Y code in user memory.

Several readers have noted that a similar set of hierarchy registers is available in the TI-95 as system registers 016 through 023 (see page C-18 of the TI-95 Programming Guide). These registers can be accessed in the unprotected mode. However, the editor believes that for most translations the user will be better served by simply selecting another data register. In future issues I will discuss some of the more complex useages for HIR commands in TI-59 programs.

Special Techniques for Saving Program Steps

When TI-59 programs became too large for the available memory the programmers used complex and obscure techniques to save program steps. One example appeared in my program "Polynomial Curve Fit with Errors" (PPX 208059). Steps 165-167 of the second stage of the program contain the sequence Pgm 15 E'. Downloading program 15 will reveal that Pgm 15 E' simply calls the sequence Pgm 01 SBR CLR to clear the statistics registers. Why go to all that trouble? I was sorely in need of an additional program step!. Note that the Master Library module must be installed anyway since ML-02 is used in the solution.

```
098  76 LBL
099  11  A
100  42 STO
101  07  07
102  38 SIN
103  42 STO
104  01  01
105  42 STO
106  06  06
107  33 X²
108  42 STO
109  02  02
110  43 RCL
111  07  07
112  39 COS
113  42 STO
114  04  04
115  49 PRD
116  06  06
117  33 X²
118  42 STO
119  05  05
120  91 R/S
```

I used an even more complex technique to save steps in another program. The upper listing at the right used 21 steps to store:

1. An input angle in R07.
2. The sine of the input angle in R02.
3. The square of the sine of the input angle in R03.
4. The cosine of the input angle in R04.
5. The square of the cosine of the input angle in R05.
6. The product of the sine and cosine of the input angle in R06.

```
098  76 LBL
099  11  A
100  42 STO
101  07  07
102  36 PGM
103  01  01
104  71 SBR
105  25 CLR
106  01  1
107  32 X:T
108  43 RCL
109  07  07
110  37 P/R
111  78 Σ+
112  91 R/S
```

The lower listing at the right achieved the same result with only 13 steps. Steps 102-105 cleared the statistics registers, that is, placed zeroes in R01 through R06. Steps 106-110 placed the sine of the input angle in the display register, and the cosine of the input angle in the t register. Step 111 summed combinations of the values in the display register and the t register into R01 through R06 according to the table on page V-33 of Personal Programming. The functions of the input angle were then recalled as needed.

Translation from the TI-59 to the TI-95 - (cont)

One could mechanize a similar technique in a TI-95 program. The program would have to recall the functions of the input angle from system registers 070 through 075 in the unprotected mode, or else recall the values using the SHOW commands. Again, I would question the use of such obscure techniques if plenty of memory were available.

These are only two examples of the kind of techniques which were used to save program steps in TI-59 programs. I don't have any good rules for recognizing such sequences or for programming the equivalent on the TI-95. Hopefully, the TI-59 programmer described the intent and the technique in the program documentation.

TRUNCATION EFFECTS IN EXTENDED PRECISION

In the errata section on page 2 Carl Rabe pointed out that the last lines of digits in several extended precision examples were incorrect. This is a direct result of the truncation that occurs beyond the last digit in the sum carried by the calculations. If all the terms of the summation are of the same sign then one might reasonably expect to find that, on the average, the truncation observed will be one half of the last digit times the number of terms in the sum. One example of such an effect occurred in Patrik Johnasson's program for calculation e to 480 digits which appeared in V9N4P24. The last line of the printout from the program is 6445490479. A 1300 digit calculation of e provided by Robert Prins in V9N5P4 will show that the correct last line for the 480 digit calculation would be 6445490598. Patrik's program required 245 iterations, so the expected truncation would be of the order of 122. If that value is added to 6445490479 the sum is 6445490601 which is within 3 of the correct last line. A nice explanation of the effect, albeit in Swedish, appears in the Programbiten article which first published Patrik's program. Send a SASE and an extra stamp if you would like a copy.

If you use the rule to check the accuracy of the solutions for Ln(2) and Ln(3) in V11N2P24 you will find that the rule doesn't seem to work. The reason is that the Ln program already provided an automatic correction to account for truncation.

A copy of the program from V11N2P24 translated to calculate Ln(3) on the TI-74 appears at the right. Lines 500-510 provide the automatic truncation correction by adding 1 at the least significant digit every other cycle, for an average of 0.5 per cycle. Without this correction the error in the last line would be substantially larger.

```
100 DIM A(120),B(120),C(
120)
110 IMAGE ##########
120 CALL UP("Logarithm o
f 3",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;" Digi
ts"
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
860 PRINT #Z
870 PRINT #Z,"I = ";I
900 IF Z=1 THEN CLOSE #1
999 STOP
```
