Sydney, New South Wales, Australia $3
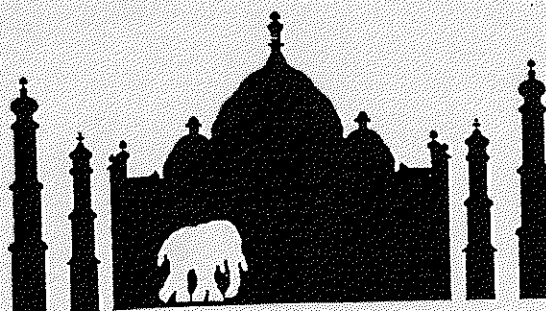
The Board

Membership and Subscriptions

TIsHUG Sydney Meeting

The November Meeting will start at
2.0 pm on the 5th November 1994
at Meadowbank Primary School,
Thistle Street, Meadowbank.

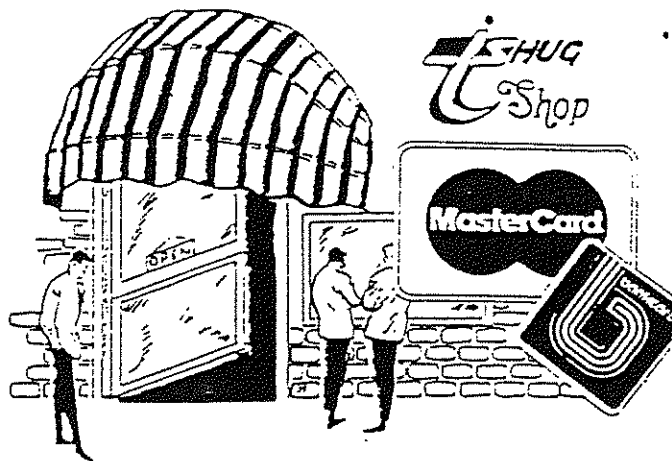TIsHUG News Digest   ISSN 0819-1984

T I   I N D E X

TIsHUG SHOP.

with Percy Harrison.

Please note that all of the 80 column cards will be complete and available at the November meeting providing you have paid for the extra cost involved of $35.00 each. For those members requiring their cards to be sent through the mail please add another $6.00 to cover cost of postage and insurance for mailing within Australia and $20 for mailing to overseas members.

We have now finalised our negotiations with a supplier of IBM compatible Hardware and Software and are pleased to advise that we are now able to supply practically all of your requirements from complete systems, through CD Roms to cables, storage boxes etc. at what we believe to be very competitive prices. Should you be able to get a better price on any item please let me know and we will endeavour to either match it or give you a slightly better price.

I have noticed that in recent weeks, prices of PC products have fluctuated quite a lot so we will not publish our prices for the more expensive items. Should you be in the market for PC systems or products please give your club the opportunity to quote you for your requirements as purchases from the club will help to keep it a viable and going concern. This offer is not restricted to club members so if you have any friends who are in the market for PC Software or Hardware please get them to contact me for prices. All products are brand new and are covered by warranty.

Following is a small sample of some of the items available through your club shop:

DESCRIPTION	PRICE
3 Button Mouse with Driver Disk	$ 14.00
Mouse Pad (Red, Grey or Blue)	$ 3.00
101 Enhanced Keyboard	$ 30.00
Screen Filter (Delux Optical 14" Monitor)	$ 20.00
Screen Filter (Delux Optical 15" Monitor)	$ 45.00
Screen Filter (Mesh 14" Monitor)	$ 8.00
Dustcover (PVC for Monitor and Desktop)	$ 8.00
Dustcover (PVC for 14" Monitor)	$ 7.00
Dustcover (PVC for 80 Column Printer)	$ 7.00
Dustcover (PVC for Mini-Tower)	$ 7.00
Copy Holder (A4 Paper)	$ 5.00
Printer Stand (for Continuous Paper)	$ 10.00
3.5 Drive Cleaning Kit	$ 6.00
5.25 Drive Cleaning Kit	$ 6.00
3.5 Diskette Mailing Box (Holds 5)	$ 2.50
5.25 Diskette Mailing Box (Holds 5)	$ 2.50
3.5 Diskette Storage Box (Holds 10)	$ 5.00
3.5 Diskette Storage Box (Holds 100 Disks)	$ 10.00
5.25 Diskette Storage Box (Holds 100 Disks)	$ 10.00
3.5 Diskette Storage Step Cube (Holds 15 Disks)	$ 7.50
Mouse Adaptor Lead (DB25F - DB9M)	$ 7.50
Mouse Adaptor Lead (DB25M - DB9F)	$ 7.50
Printer Cable 6ft (DB25 - C36M)	$ 6.00
Printer Cable 6ft (DB25 - C36M)	$ 6.00
Printer Cable 10ft (DB25 - C36m)	$ 9.00
Printer Cable 15ft (DB25 - C36m)	$ 9.00
Printer Cable 25ft (DB25 - C36M)	$ 19.00
Serial Cable RS232 6ft (DB25M - DB25M)	$ 10.00
Serial Cable RS232 10ft (DB25M - DB25M)	$ 12.00
Serial Cable RS232 15ft (DB25M - DB25M)	$ 15.00
Serial Cable RS232 25ft (DB25M - DB25M)	$ 20.00
Laplink Cable 6ft (DB25M - DB25M)	$ 22.00
Laplink Cable 15ft (DB25M - DB25M)	$ 28.00
Modem Cable 6ft (DB25M - DB25F)	$ 9.00
Modem Cable 6ft (DB25M - DB9F)	$ 8.00
VGA Monitor Extension Cable	$ 9.00
Keyboard Extension Cable	$ 6.00
Power Split Cable for 3.5 Drives	$ 6.00
Power Split Cable for 5.25 Drives	$ 6.00
PC Tool Kit (13 Piece)	$ 28.00
Creative Lab 16 Bit Sound Card with Speakers	$199.00
Panasonic Double Speed CD Rom Drive	$240.00
Discovery Kit	$530.00
1MB SIMMS x 70ns RAM	$ 63.00
4MB SIMMS x 70ns RAM	$238.00

Note: 1MB SIMMS in quantities of 4 at $60.00 per MB.

As I mentioned above this is only a sample of what we can supply. If you require any of the following please contact me:

CD ROM Drives
Commercial Software
Complete Systems, 386 or 486 (SX or DX)
Floppy Disk Drives 1.2 and 1.44
Hard Disk Drives and Controllers
Math Co-Processors

Modems
Monitors 14", 15" and 17"
Motherboards
Printers (Dot Matrix, Inkjet and Laserjet)
Sound Cards
Tape Backup Units
Tower Cases
VGA Cards

Remember, your club will only grow and maintain it's services to you if you continue to support it so please give us the opportunity of quoting you for you computer requirements. If we can't match or better the price that you get from other suppliers then by all means buy where you save yourself money.

Bye for now.

END OF ARTICLE

DUMPING DATA INTO MULTIPLAN

How To Use The SYLK File Format
Part 1
by Bob Relyea

INTRODUCTION AND OVERVIEW

I have always felt that a drawback to the TI system of Funnelweb/Multiplan/TI-Base was the lack of access of Multiplan to data generated by, for example, TI-Base or a Basic/Extended Basic program. Multiplan is able to do things with data that other TI programs are not capable of, so to be able to 'dump' data generated by other programs into Multiplan for final manipulation would be a definite advantage to our system.

I had thought about this problem on and off for years, but until recently I had never gotten anywhere. A few years ago, when people were still taking their TI's seriously, I enquired into this problem and I was told that there was a provision for doing this in the original software using the SYLK file format as described in appendix 4 of the Multiplan manual. Rolf dug up a couple of articles printed years ago in MICROPENDIUM that served as an introduction for me to the use of SYLK files. This article will be the first in a series to explain what has transpired between the reading of those first two articles in MICROPENDIUM and now.

The TI system is so good and planned with such ingenuity that after all these years we are still learning about its capabilities and having fun doing things with it. I find that using the more recent 'click-click' machines (i.e. Macs & PC's) is next to drudgery and little fun at all. Most people buy them because they have to 'keep up with the Jones' and in reality do not do anymore with their machines than we normally do with the TI. So, why spend the money?

Anyway, back to the main point of this article. I read through the two articles that Rolf gave me (December, '86 and January, '87), had a go at typing in both programs and got them both to work. Beyond that, I did not get very far on my first attempt, as I found the articles hard to understand. The problem was greatly complicated by the fact that the second article was poorly written. I noticed only this year that the author had written the article based on an earlier version of his program and after the article was written he apparently RESequenced the program so that the line numbers in the program did not match the description in the article. This certainly did not help my introduction to SYLK files! I have never noticed anywhere in an Australian magazine any substantial articles on the topic of SYLK files. After taking my first look at appendix 4 where they are described I can see why! Appendix 4 is written in typical computer jargon and especially suited to the technically-minded person who is used to reading things designed for robots. A lot of valuable information is condensed in the short space of a few pages. Because the two programs in MICROPENDIUM did not really suit my needs and I could see no way of adapting them for a more universal application I gave up on the idea for a while. I think that over a period of three years that I picked up those articles several times along with appendix 4 and the manual in an attempt to get on top of the matter but to no avail. Just this year I felt the need to have another look as I was generating data in an Extended Basic program that I had written that I really needed to dump into Multiplan in order to make life simple. So, I dug out the SYLK file gear, blew the dust off and said "I am going to get on top of this SYLK file format if it is the last thing that I do!"

You may remember reading past articles in the TND about a Grade Standardising program that I had been working on and using for my teaching. I greatly expanded the original version to allow me to do a whole range of things, such as printing out a hard copy, cataloging a disk, etc. However, in high school teaching these days students in years 10 and 12 have to do assessment tasks. This means that there are, say, six assessable tasks that have to do during the course of the year, each with a certain weighting, which give them their final ranking amongst all the other students

in the form. So, with my Extended Basic program, I feed in the marks from each assessment task one by one as they do them over the course of the year, standardise each set, print out a hard copy for my use and then store the data on disk. The problem is that at the end of the year all of these grades from the six assessment tasks have to be assessmbled together, each with their weighting, to get a final grade. My program is not capable of doing this but Multiplan is. So, I really needed a way of dumping the data from these individual files onto a single spreadsheet so that Multiplan could finish the job. This is when I got desperate to get on top of the Sylk file problem. At one of the Illawarra regional meetings Geoff and I had another look at the problem and I got enough new clues and encouragement to take the matter a few steps further. Geoff outlined a program which served for me to flesh out and use as a basis for further study and experiment. I have now solved my original problem and have gotten to the stage in my understanding to be able to help other people who are interested in this very valuable feature of Multiplan. The next part of this article will start to explain the basics of using the SYLK file format and how it can be adapted for other uses. If interested, read on!

FIRST PRINCIPLES

To start with you must read and re-read Appendix 4 of the Multiplan manual, pages 205 to 208. This is not a lot of reading but a lot of information is condensed into a very short space. It will not be easy to understand it all the first or second (or third?) time through but you must start somewhere. I suggest that you make a photocopy of these pages and also a photocopy of pages 132 to 136 as these pages help to explain the parameters mentioned in the appendix. On your photocopied pages you can make notes and shuffle them around the way you like. Do not be put off by the comment in the introduction on page 205 that "you should only use the manipulations made possible by the Symbolic Link if you have an advanced understanding of the detailed structure of ASCII files". I am not sure that my understanding is "advanced" and I made out OK, so if you do have an advanced understanding, then all the better! The big drawback of this section is that no examples are given. There is a provision, however, for you to make your own examples. When you get to the stage where you want to see some examples you boot up Multiplan, select Transfer, Options and then (S)ymbolic. This means that while you are on this setting you can only Save or Load in the Symbolic mode. You then make up a Multiplan file and save it to disk using the Symbolic File Format and then call the file up on a sector editor and have a look at what it did. Pay special attention to the beginning and end of it as that is where the essential parts of the symbolic format lie. This will give you valuable information on how to write your own program later on to produce the exact Multiplan file that you want. More on this later. Another thing

to do in this beginning stage is to read the two MICROPENDIUM articles written by Bill Harms that I referred to before which are printed at the end of this article for convenience. Like we did in the early years of the club when we were having fun with the TI, it pays to actually type the programs in and run them to see what it does. A lot sticks with you by just typing up somebody elses article and having a go at it. All you really need to do is to slightly modify the parameters given in lines 200 of his first article and lines 230 - 232 and line 350 of the second article and from there it is really a matter of writing a program to suit your needs. Above all it will be necessary to read these articles through several times in order to really absorb it. I think I will let it go at that for this article and in the next however, for you to make your own examples. When you get to the stage where you want to see some examples you boot up Multiplan, select Transfer, Options and then (S)ymbolic. This means that while you are on this setting you can only Save or Load in the Symbolic mode. You then make up a Multiplan file and save it to disk using the Symbolic File Format and then call the file up on a sector editor and have a look at what it did. Pay special attention to the beginning and end of it as that is where the essential parts of the symbolic format lie. This will give you valuable information on how to write your own program later on to produce the exact Multiplan file that you want. More on this later. Another thing to do in this beginning stage is to read the two MICROPENDIUM articles written by Bill Harms that I referred to before which are printed at the end of this article for convenience. Like we did in the early years of the club when we were having fun with the TI, it pays to actually type the programs in and run them to see what it does. A lot sticks with you by just typing up somebody elses article and having a go at it. All you really need to do is to slightly modify the parameters given in lines 200 of his first article and lines 230 - 232 and line 350 of the second article and from there it is really a matter of writing a program to suit your needs. Above all it will be necessary to read these articles through several times in order to really absorb it. I think I will let it go at that for this article and in the next issue I will start to explain and make clear the symbolics in the abovementioned lines. Here, in very slightly modified form, are Bill Harms' articles.

WORKING WITH MULTIPLAN
by Bill Harms
MICROPENDIUM, December, '86
Retyped/Edited by Bob Relyea

In this article I will introduce you to a method to transfer data from a BASIC program for use by Microsoft Multiplan.

I use Multiplan to keep my budget and to estimate my income taxes. I have a spreadsheet with 18 columns: 12 months, Yearly Total, Year-To-Date, Weekly Average, Monthly Average and two for taxes. Those last two have formulas to get various numbers from the spreadsheet. The rows include the following categories: Pay, Interest, Expenses, Loans and Other. You can really do "What If'ing" and "Why Not'ing" with Multiplan.

It was a pain, however, adding up all the darn monthly checkbook entries in separate categories (Meals Out, My Pay, My Wife's Pay, Groceries, Interest, etc).

Since Multiplan does not have a FOR/NEXT capability and cannot scan data until some criteria is met it was too time-consuming to enter the 100 to 150 monthly entries directly from my checkbook into Multiplan, even if there was enough RAM.

Now I use a fast little Extended Basic program I wrote to get those transactions added by category. Then I can use a SYLK creator to quickly and correctly prepare them for loading into my Multiplan spreadsheet.

SYLK (or Symbolic Link) files are a little known feature of Multiplan. They can be written to disk by a BASIC program and read by Multiplan.

In this article I will show how this is done. This material is based on a program I got from Texas Instruments, a series of articles in the May (and later) 1895 Super 99 Monthly (now called the Smart Programmer) and the Multiplan manual.

This bare-bones program is based on the one I received from TI in 1984. The disclaimer was bigger than the program! It writes a disk file with a one cell spreadsheet (do not worry about the size, bigger things are to come! ED) that can be read by Multiplan.

```
100 OPEN #1:"DSK1.SYLKF",DISPLAY ,OUTPUT,FIXED 128
110 CALL CLEAR
120 INPUT "ROW NUMBER: ":R$
130 INPUT "COLUMN NUMBER: ":C$
140 INPUT "CELL CONTENT: ":A$
150 FOR Q=1 TO 27-LEN(A$)
160 W$=W$&CHR$(0)
170 NEXT Q
180 X$=CHR$(34)&A$&CHR$(34)!SURROUNDS CONTENTS WITH
    QUOTES
190 Z$=CHR$(13)&CHR$(10)! CARRIAGE RETURN AND LINEFEED
200 Y$="ID;PMP"&Z$&"F;DGOGB"&Z$&"B;Y"&R$&";X"&C$&Z$&"C;

    K"&X$&Z$&"W:N1;A1 1"&Z$&"
    E"&Z$&W$
210 PRINT #1:Y$
220 CLOSE #1
230 END
```

Line 200 has the symbolics needed for Multiplan to read the file. See page 205 of the Multipan manual for explanations.

If you enter and run this program, you will find a file on your disk called "SYLKF". Before you can load this file, you must change it. It may seem a bit odd, but the file must be written as DISPLAY, FIXED 128 and then changed to INTERNAL, FIXED 128 in the file header. In other words, the file must use DISPLAY notation but

There are two ways to do this. One way is to use a sector editor to change the last four hex characters of the first line (in a 40 column version) of the file header to 0202. (The other way is to program it in to be done automatically- the subject of a future article. ED)

Once you have done this you may load your file. First, boot Multiplan, Press (T)ransfer and then (O)ptions. Next press (S)ymbolic and then ENTER. Now press (T)ransfer again and this time (L)oad your file:

```
|  ---------  |
|             |
|    HARMS    |
|             |
|  ---------  |
```

This is what the data looks like on a sector editor. It presumes that you have chosen 1 for row, 1 for column and 'HARMS' for cell content when prompted. Y1 and X1 stand for row 1, column 1.

```
I D ; P M P * * F ; D G O G
B * * B ; Y 1 ; X 1 * * C ;
K " H A R M S " * * W ; N 1
; A 1   1 * * E * * * * * *
```

Most of the the asterisks stand for CR/LF (carriage return/linefeed), Z$ in line 190.

There are many ways you could input data besides the simple INPUT in line 140. You could read data in from DATA statements or from a disk file. That disk file could be created by most anything: TI-Writer, RS232, another module or a Multiplan Print File.

You can create data in Basic and then "dump" it into a spreadsheet en masse instead of just keyboarding it.

You could transmit the outputted SYLK file of your Multiplan spreadsheet to others via the RS232. The DIF (Data Interchange Format) used by Lotus 1-2-3 and Visicalc only accomodates the cell content, not the sheet parameters.

This is only a taste of what you can create to load data into Multiplan. It really opens Multiplan up to other software.

Next month we'll do a 2 row by 2 column spreadsheet. —

MULTIPLAN MACHINATIONS
by Bill Harms
MICROPENDIUM, January, '87
Retyped/Edited by Bob Relyea

This is the second of two articles describing how to use Basic/Extended programs to create Symbolic Link (SYLK) files to load data in Microsoft Multiplan-Ed.

The first article showed how to create a Multiplan SYLK file for one cell. It described only a few of the many SYLK symbols. The most significant change in this month's program is that it can create a file with many cells- a couple of columns of data.

After using this program, you could load the file into Multiplan and save it as a normal file. Then you could eXternal Copy it or parts of it into another spreadsheet that has formulas and other data. You might load a month's worth of information into a sheet that has many months of data, plus formulas for calculating Year-to Date, Average, Units per Time Period, etc.

As I mentioned last month, the BASIC program must write a Display Fixed 128 file. Otherwise, Multiplan cannot read it. As described in the first article, you must use a sector editor to change the last four numbers of the header from 0002 to 0202.

Each SYLK record (files are made up of records) can include cell content (text, numeric value or formula), row and/or column numbers, and many other symbols to describe windows, sheet boundaries, formatting of sheet and individual cells, sheet links and more. In fact, just about everything may be included, except commands that Multiplan uses such as COPY and DELETE.

Your cell content data may be split into more than one record. You just keep on creating 128 character records of SYLK symbols and data until you run out of infrmation that you want to put into the SYLK file with nulls-CHR$(0)-so that it is 128 characters long.

Last month's program had 27 nulls, which made it a 73-byte record. I could not get it to load using a larger number of nulls (such as 123). I found the key in Richard Mitchell's program, published in the May, June and August, 1985 Super 99 Monthly. You continue building a string (numerics are included via STR$(xx)), using the ampersand to concatenate each new symbol or data item onto the string as you go until the LENgth of the string exceeds 128 characters of the string into your disk SYLK file. Right after printing you move any characters in your string beyond 128 to a temporary variable.

Then you move the remaining characters back into your main string and continue building until it has more than 128 characters again, and you do another print to disk of another SYLK record (see WRITE sub program below). There are two types of SYLK symbols: Record Type Descriptors (RTD) of 1 or 2 characters; and Field Type Descriptors (FTD) which are preceeded by a semicolon. These need to be surrounded by double quote marks (not mentioned in te Multiplan manual). Carriage returns and linefeeds are used to separate SYLK record types- a Record Type or RTD is a SYLK symbol while a record is part of a file. Records contain record types.

Let's look at a sample: "C;K". This is an RTD C, which means that it is a data point (i.e. what goes into a cell). The FTR ;K means that the value of the data point follows. Once you set a row or column number with the RTD C and FTD of, say, ;Y (for row), all the remaining data points (;K or ;E) are put at that row. You only need another "C;X" or "C;Y" when changing the row (;Y) or column (;X) of the data point.

One of the fastest ways to learn the correct formatting of SYLK records is to enter some data into Multiplan and save the sheet in Symbolic format. Then you look at the fule with a sector editor.

The following program creates a SYLK file of 2 columns with 4 rows of values and a formula for the sum of the second column. The second column is NAMEd per your choice.

NAMEing is a Multiplan technique. It is quite helpful, since a name of a cell or a range of cells can be used in a formula; to wit, SALES-COST (SALES minus COST). You would have NAMEd some cells "SALES" and some others, "COST", so the cell with this formula yields the profit. It could be named "PROFIT".

A normal formula might look like R4C6-R9C6. For a bunch of data you might use SUM(R1:40C3)+SUM(R90C4:5). Relative references (relative to the cell that has the formula), as in my program, look like SUM(R[-1]C:R[-2]C). The whole sheet looks like this when loaded into Multiplan:

	1	2
1	18	15
2	1	75
3	33	199
4	400	77
5		------
6		366

The value in R6C2 (366) is from the formula SUM(R[-5]C:R[-2]C). Using a sector editor we can see the whole sheet!

```
I D ; P M P * * F ; W 1   2
  6 * * F ; D G 2 G B * * B
  ; Y 7 ; X 3 * * N N ; N W H
  H ; E R 1 : 6 C 2 * * C ; Y
  6 ; X 2 ; E S U M ( R [ - 5
  ] C : R [ - 2 ] C ) * * C ;
  X 1 C ; Y 1 ; K 1 8 * * C ;
  Y 2 ; K 1 * * C ; Y 3 ; K 3
  3 * * C ; Y 4 ; K 4 0 0 * *
  C ; X 2 C ; Y 1 ; K 1 5 * *
  C ; Y 2 ; K 7 5 * * C ; Y 3
  ; K 1 9 9 * * C ; Y 4 ; K 7
  7 * * C ; Y 5 ; K " - - - -
  - - " * * W ; N 1 ; A 1   1
  * * E * * * * * * * * * * *
```

The 128 Character SYLK record ends after the fourth character in the 15th line. This is also the end of the SYLK file.

Mitchell's program reads a Display Variable 80 file and writes the data into a SYLK file that Multiplan can read. It has one word processor line on each row with five nice columns. You could adapt it to several uses.

Multiplan has a 255-row and 64-column limit. I have found that about 23 columns and 40 rows of formulas, data and labels (text) is about all that the 4A's RAM can hold.

The program I wrote, which is based on Mitchell's, follows. These comments may help you to understand it.

```
100 ! ON DISK: SYLK/24
110 DISPLAY AT(1,5)ERASE ALL:"A SMALL SPREADSHEET"
120 ! DATA INPUT SECTION **
130 OPTION BASE 1
140 DIM AMT(4,2)! ARRAY FOR INPUT DATA **
150 DATA 18,15
160 DATA 1,75
170 DATA 33,199
180 DATA 400,77
190 FOR A=1 TO 4 !ROWS
200 FOR B=1 TO 2 !COLUMNS
210 READ AMT(A,B)
220 NEXT B :: NEXT A
230 ! END OF INPUT ROUTINE ***
240 !
250 DISPLAY AT(6,1):"ENTER DESIRED SYLK FILE NAME  DSK"
260 ACCEPT AT(7,3)BEEP SIZE(-15):FILE2$ :: IF FILE2$=""
    THEN 260
270 OPEN #2:FILE2$,DISPLAY ,FIXED 128,OUTPUT
280 DISPLAY AT(9,1):"ENTER DESIRED NAME FOR 2nd COLUMN."
    :: ACCEPT AT(10,10)SIZE(10)VALIDATE(UALPHA):NAME$ ::
    IF NAME$="" THEN 280
290 R$=CHR$(13)&CHR$(10)
300 T$="ID;PMP"&R$&"F;W1 2 6"&R$&"B;Y"&STR$(A+2)&";X"&
    STR$(B)&R$
310 T$=T$&"MN;N"&NAME$&";ER1:6C2"&R$
320 T$=T$&"C;Y6;X2"&";ESUM(R[-5]C:R[-2]C)"&R$
330 !
340 FOR COL=1 TO 2
350 T$=T$&"C;X"&STR$(COL)
360 FOR ROW=1 TO A-1
370 IF LEN(T$)>128 THEN CALL WRITE(T$,T1$):: T$=T1$
380 T$=T$&"C"
390 T$=T$&";Y"&STR$(ROW)
400 T$=T$&";K"&STR$(AMT(ROW,COL))&R$
410 NEXT ROW
420 NEXT COL
430 !
440 T$=T$&"C;Y"&STR$(ROW)&";K"&CHR$(34)&"------"&CHR$
    (34)&R$
450 IF LEN(T$)>128 THEN CALL WRITE(T$,T1$):: T$=T1$
460 T$=T$&"W;N1;A1 1"&R$&"E"&R$
470 IF LEN(T$)>128 THEN CALL WRITE(T$,T1$):: T$=T1$
480 PRINT #2:T$&RPT$(CHR$(0),128-LEN(T$))
490 CLOSE #2
500 SUB WRITE(T$,T1$)
510 PRINT #2:SEGS(T$,1,128)
520 T1$=SEGS(T$,129,LEN(T$)-128)
530 SUBEND
```

Line 300 adds sheet identification, windows, format and bounds.

Line 310 adds the name you inputted in line 280 and the range for that name (R1:6C2).

Line 320 adds the formula for R6C2.

line 400 adds the cell content as numerics (here taken from the array AMT).

Lines 370, 450 and 470 work together to ensure that each record is 128 characters long.

Lines 500 through 530 write the SYLK records to disk.

Rather than using the DATA statements you may decide to use a file INPUT statement, in which case you would not use the READ statement in line 210. Remember, that after running this program, you will need to change the FDR to 0202 using a sector editor. And, before you attempt to load this SYLK file into Multiplan, make sure to select the Symbolic option from the Multiplan (T)ransfer menu. Now you can load your data into Multiplan to suit your needs.

See you next month for Part 2 where we start having a serious look at the symbols needed to put a SYLK file together. Happy computing.

```
 _____
| END OF ARTICLE  >>>>>>>|
 ---------------------------
```


TREASURER'S REPORT

by Cyril Bohlsen

Income for previous month .... $ 1109.00

Expenditure for previous month .. $ 272.41

Profit for previous month .... $ 836.18

Membership accounted for $ 35.00 of Income.

Shop sales .... $1074.00 of Income.

The expenditure was made up of the following

Printing & Postage of TND .... $ 272.41

Notice to Members

All members are advised that the 8th Annual General Meeting of TIsHUG (Australia) Limited will be held on Saturday, 3rd December, 1994 at Meadowbank Primary School, Thistle Street, Meadowbank NSW, commencing at 2.00pm.

Members attending are requested to arrive by 1.30pm to enable them to sign in and to ensure a prompt 2.00pm start.

The following relevant paragraphs from the club's Articles of Association are brought to your attention:

16(i) - Nomination for the office of Director shall be delivered to the Secretary by 8.00pm on the twenty-first day prior to the day fixed by the board for the Annual General Meeting.

17(b) - Nominations for election of the Directors shall be made in writing and signed by two (2) members of the club and by the nominee who shall signify his consent to the nomination.

17(d) - If the full number of candidates for the positions of Directors is not nominated as prescribed then additional nominations may be made at the meeting. If there be more than the required number nominated an election by ballot shall take place but if there be only the requisite number nominated the Chairman shall declare those nominated duly elected.

In accordance with paragraph 16(i), nominations for the office of Director shall close with the Secretary at 8.00pm on Saturday, 12th November, 1994, while in accordance with paragraph 17(b) a suitable nomination form is enclosed.

Percy Harrison (for Honorary Secretary)

TIsHUG (Australia) Limited
EIGTH ANNUAL GENERAL MEETING
Saturday 3rd December, 1994
Meadowbank Primary School
Thistle Street Meadowbank NSW

Agenda

1. Meeting opening.

2. Members Present and apologies.

3. Reading and confirmation of minutes of the 7th Annual General Meeting held on Saturday 4th December 1993.

4. Correspondence and dealing with same.

5. Co-ordinator's Report.

6. Treasurer's Report, Accounts and Auditor's Report.

7. Unfinished Business from last AGM (if any).

8. Election of Returning Officer and two (2) Scrutineers.

9. Election of Directors.

10. Election of Auditor.

11. New Business (if any).

12. Close Meeting.

TRANSLITERATE TIPS

By Jim Peterson
Retyped by Loren West

We all know that the TI-Writer formatter insist on giving us five blank lines at the top of the page and three at the bottom. If your printer supports a reverse line feed, you can back those lines at the top by beginning the page with a line CTRL-U and RJRJRJRJRJ; but I can't find a practical way to print the three lines at the bottom.
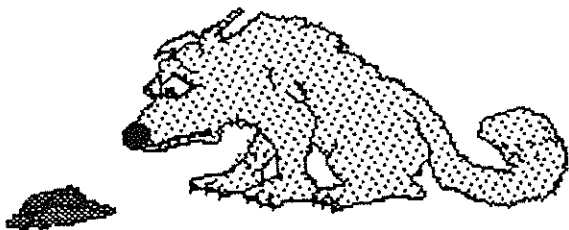
MINUTES OF AGM

TIsHUG (Australia) Limited
MINUTES OF AGM
4-12-93
Meadowbank Primary School
Convened at 2:40 pm

Number of members present- 29

Directors Present- Richard Warburton, Cyril Bohlson, Percy Harrison, Peter Schubert
Absent- Russell Welham

1. The Coordinator, Richard Warburton called the meeting to order.
   Apologies- John Robinson, Ashley Lynn, Alf Culloden, Derek Wilkinson, Vincent Maker, Larry Saunders, Daniel Harris, Mike Slattery.

2. The minutes of the previous AGM (5-12-92) were read.
   Percy Harrison moved that they be accepted.
   Peter Young seconded the motion.
   Carried.

3. Business arising from the previous minutes- nil

   4. Correspondence relating to the AGM of 5/12/92- nil

5. Life Membership

   Percy Harrison moved that- "as from 4-12-93 the club does not nominate nor confer any new life membership on any new member until the financial situation, as seen by the directors, warrents it."
   Ian Mullins seconded the motion.

   The motion was briefly discussed and the names of the present life members was given- Shane ANDERSEN, Cyril BOHLSON, Ross MUDIE, Terry PHILLIPS, and Geoff TROTT.
   The motion was carried.

6. Director's Reports

   This reports were circulated to the members weeks prior to the AGM for perusal. The Coordinator, Dick Warburton thanked the Treasurer, Cyril Bohlson for his effort in producing the financial report. Percy Harrison pointed out that in the financial statement of the previous year the TIM cards were not taken into consideration. In this year's report they were which explains why we have incurred a loss. No further discussion was forthcoming so-
   Don Gould moved that the Director's Reports be accepted.
   Tom Marshall seconded the motion.
   Carried.

7. Election of Officers

   All directors positions were declared vacant.
   Those standing for the five positions of director were: Cyril Bohlson, Percy Harrison, Vincent Maker, Tom Marshall, Bob Relyea and Dick Warburton.
   Ross Mudie and Geoff Trott were selected to be scrutineers.
   Those elected were: Cyril Bohlson, Percy Harrison, Tom Marshall, Bob Relyea and Dick Warburton.

8. Auditor

   Percy Harrison moved that the present auditor be retained.
   Alf Ruggeri seconded the motion.
   Carried.

9. New Business

   A vote of thanks was given to the ladies for the good job with the BBQ. It was suggest that they be put in charge of next years BBQ and be told about it well in advance.

10. Since there was no further business the meeting was closed by Dick Warburton at 3:10pm.

LESSON 21

LEARN TO KNOW YOUR TI

with Percy Harrison

Well how did you make out with mastering sound or noise on your computer last month? I hope that it was not too difficult and by now you have written some sound programs for yourself. Instead of giving you the answers for the sounds that I asked you to produce in your Assignment I thought that I would include a few different ones for you to take a good look at and enter on your TI and run then then so that you will get a better understanding of the sound features that are available to you on your TI. By the way if any of you have produced the sounds that I asked for in the Assignment please send them to me and I will pick the best ones out and publish them in our magazine.

This month we will take a look at the HI-RES Graphic capabilities of your TI machine.

TI BASIC has a powerful command, CALL CHAR(), to define new characters.

It can be used to produce a new type font or make game characters. For example, 8 arrows, cars, planes, tanks,etc., pointing in different directions allow plenty of action in games. You can make ships, dinosaurs, and other elaborate high resolution graphics by using 2 or more adjacent characters.

In doing this lesson you will need to use a grid comprising two groups of 8 boxes deep and 4 boxes wide , placed side by side forming a total grid of 64 boxes ie: 8x8. There is a grid in this lesson that you can reproduce by office copier (when your boss is not looking) so you will have grids to practice on without defacing your magazine.

LESSON 21 HI-RES GRAPHICS

Learn to make your own graphics characters for drawing "high-resolution" pictures.

REVIEW OF GRAPHICS

Let us review the graphics you have already learned in previous lessons.

THE COLOURS

1	TRANSPARENT	9	MEDIUM RED
2	BLACK	10	LIGHT RED
3	MEDIUM GREEN	11	DARK YELLOW
4	LIGHT GREEN	12	LIGHT YELLOW
5	DARK BLUE	13	DARK GREEN
6	LIGHT BLUE	14	MAGENTA (PURPLE)
7	DARK RED	15	GREY
8	CYAN (BLUE-GREEN)	16	WHITE

Adjust your TV controls to make the colours look right.

THE CHARACTER SETS

There are 16 sets. Each has 8 characters in it.

Set	Character Number							
1	32	33	34	35	36	37	38	39
2	40	41	42	43	44	45	46	47
3	48	49	50	51	52	53	54	55
4	56	57	58	59	60	61	62	63
5	64	65	66	67	68	69	70	71
6	72	73	74	75	76	77	78	79
7	80	81	82	83	84	85	86	87
8	88	89	90	91	92	93	94	95
9	96	97	98	99	100	101	102	103
10	104	105	106	107	108	109	110	111
11	112	113	114	115	116	117	118	119
12	120	121	122	123	124	125	126	127
13	128	129	130	131	132	133	134	135
14	136	137	138	139	140	141	142	143
15	144	145	146	147	147	149	150	151
16	152	153	154	155	156	157	158	159

THE COMMANDS

Colour a character set using the CALL COLOR statement:

Enter: 10 REM GRAPHICS
         15 CALL CLEAR
         20 CALL COLOR(2,7,15)

This paints all 8 characters in set 2. They become red with a grey background.

Set 2 looks like this now,

Number	40	41	42	43	44	45	56	47
Character	(	)	*	+	,	-	.	/

but we will soon change some of the characters to look different. You put characters on the screen with CALL HCHAR and CALL VCHAR.

Add: 30 CALL HCHAR(3,6,42,4)
        99 GOTO 99

Run the program. Line 30 means:

        30 CALL HCHAR (row 3,col.6,char.42,repeat
                          4 times)

MAKING A NEW CHARACTER

Each character is a square made of dots. The square is
8 dots across and 8 dots down.

Here is an arrow character:

	0	8
	1	C
	2	A
	0	8
	0	8
	0	8
	0	8
	0	0

Figure 1

You have to tell the computer two things:

        which character number to use
        which dots to light up

Let's change the star character to an arrow:

Add: 40 CALL CHAR(42,"081C2A0808080800")
        41 CALL VCHAR(10,11,42,8)

Run the program.

When line 40 is executed, character 42, which used to be
the star "*", becomes the arrow.

IMPORTANT! Even the stars already on the screen change
            to arrows!

Add: 35 FOR T=1 TO 400
        36 NEXT T

And run the program again. Watch to see the "old" stars
change to arrows.

CODE FOR THE CHARACTER

A special code tells the computer which dots are in the
character.

Each character is divided up into 16 little rows of 4
dots each.   Each kind of 4-dot row has its own code
name,which is determined by which particular dot or dots
in the row are turned "on" or "off". The code is a
number or letter: 0 1 2 3 4 5 6 7 8 9 A B C D E F.

Figure 2

(If you know and understand binary and hexadecimal
numbers, you can see why each little row is named the
way it is. If you don't understand binary numbers etc.
it doesn't really matter so don't despair.

THE REST OF THE ARROWS

Here is a tilted arrow:

0	0
7	8
6	0
5	0
4	8
0	4
0	2
0	0
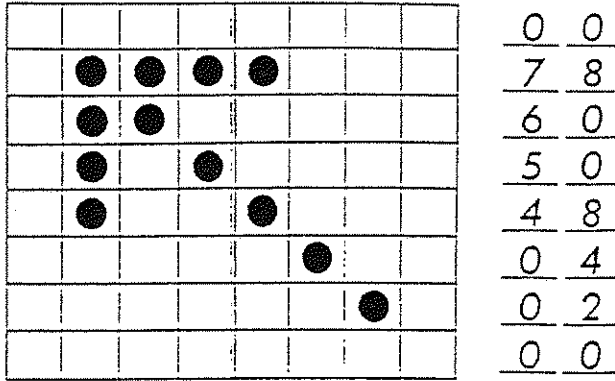
Figure 3

THE TRANSPARENT COLOUR

```
Run:    10 REM FLASHING BACKGROUND
        15 CALL CLEAR
        20 CALL CHAR(42,"081C2A0808080800")
        30 CALL COLOR(2,7,15)
        31 CALL HCHAR(12,14,42)
        35 FOR T=1 TO 400
        36 NEXT T
        40 CALL COLOR(2,7,1)
        45 FOR T=1 TO 400
        46 NEXT T
        50 GOTO 30
```

The arrow sits in a little square that flashes white, then "transparent". (The green background shows through).

Line 20 makes an arrow character and stores it in the memory.
Line 30 colours it red with a grey background.

        30 CALL COLOR(char.42,red colour,white background)

Line 40 colours it red with a "transparent" background.

Change: 40 CALL COLOR(2,1,15)

Run it again. What happens?
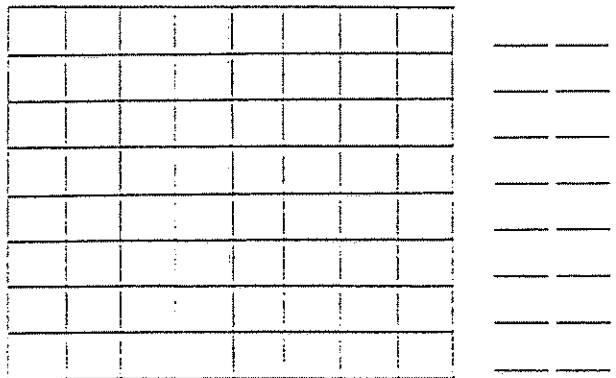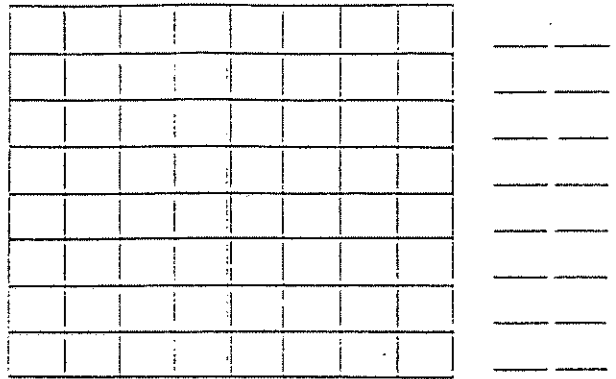
Figure 4

Assignment 21:

1. Draw 6 more arrows, so you have 8. One arrow in each direction: up,down,left,right,and four tilted arrows in between the others.

2. For each arrow, write its code letters

3. Add the above arrow characters to the above program, using CALL CHAR() command lines. Use all the numbers 40 to 48 which are in character 2 set.

4. Add lines to the program so all the arrows on the screen revolve like a little clock hands. Use CALL HCHAR and CALL VCHAR in a loop which has a delay loop.

5. Now put the 8 arrows in 8 different character sets. Make each set a different colour. When the arrows slowly revolve, they will also change colour!

ANSWERS TO LESSON 20

Assignment Question 20-2

Note, these are not the sounds that I asked you to produce, see the opening paragraphs for an explanation why.

```
10 REM SOUND PROGRAM
20 CALL CLEAR
30 CALL SCREEN(7)
40 RANDOMIZE
50 GOSUB 980
60 LET Z$="123456789ABC"
70 CALL KEY(3,K,S)
80 IF S<1 THEN 70
90 LET F$=CHR$(K)
100 FOR I=1 TO 12
110 IF F$=SEG$(Z$,I,1) THEN 140
120 NEXT I
130 GOTO 70
140 LET X=I
150 ON X GOSUB 170,220,280,360,420,490,560,610,670,
    740,820,920
160 GOTO 70
170 REM FALLING SOUND
180 FOR I=1 TO 30
190 CALL SOUND(100,2000-50*I,0)
200 NEXT I
210 RETURN
220 REM BOUNCE
230 FOR I=1 TO 5
240 CALL SOUND(100,110,0)
250 CALL SOUND(100,440,0)
260 NEXT I
270 RETURN
280 REM WAVES
290 FOR I=1 TO 2
300 CALL SOUND(400+INT(200*RND),-7,0)
310 CALL SOUND(2000,-6,5)
320 CALL SOUND(4000,-5,12)
330 CALL SOUND(INT(200*RND),4400,15)
340 NEXT I
350 RETURN
360 REM ALERT
370 FOR I=1 TO 8
380 CALL SOUND(200,440,0,400,0)
390 CALL SOUND(200,880,0,800,0)
400 NEXT I
410 RETURN
420 REM URGENT RUNNING
430 FOR I=1 TO 30
440 CALL SOUND(10,440,0,450,0)
450 CALL SOUND(10,450,0,460,0)
460 CALL SOUND(10,460,0,470,0)
470 NEXT I
480 RETURN
490 REM UFO MOVING
500 FOR J=1 TO 8
510 FOR I=1 TO 7
520 CALL SOUND(100,-1,I,2000,0)
530 NEXT I
540 NEXT J
550 RETURN
560 REM DEATH RAY
570 FOR I=1 TO 8
580 CALL SOUND(100,-3,0,880,0,890,0)
590 NEXT I
600 RETURN
610 REM CRASH
620 CALL SOUND(200,-5,0)
630 FOR I=1 TO 3
640 CALL SOUND(300*I,-7,I*5)
650 NEXT I
660 RETURN
670 REM BLAST OFF
680 FOR I=1 TO 15
690 CALL SOUND(50,-5,4)
700 CALL SOUND(10,900+30*I,0)
710 NEXT I
720 CALL SOUND(2000,-7,0)
730 RETURN
740 REM CLOCK BELL
750 FOR L=1 TO 6
760 FOR L2=0 TO 30 STEP 3
770 CALL SOUND(-50,400,L2,2400,L2)
780 NEXT L2
790 FOR D=1 TO 200 :: NEXT D
800 NEXT L
810 RETURN
820 REM AMERICAN SIREN
830 FOR T=1 TO 3
840 FOR L=800 TO 1200 STEP 8
850 CALL SOUND(-60,L,0)
860 NEXT L
870 FOR L=1200 TO 800 STEP -8
880 CALL SOUND(-60,L,0)
890 NEXT L
900 NEXT T
910 RETURN
920 REM EUROPEAN SIREN
930 FOR L=1 TO 4
940 CALL SOUND(400,500,0)
950 CALL SOUND(400,300,0)
960 NEXT L
970 RETURN
980 REM TITLES
990 PRINT "LIBRARY OF SOUNDS"
1000 PRINT
1010 PRINT "TO HEAR A SOUND"
1020 PRINT "PRESS THE APPROPRIATE KEY"
1030 PRINT "(ALPHA LOCK DOWN)"
1040 PRINT
1050 PRINT "1 -FALLING SOUND"
1060 PRINT "2 -BOUNCE"
1070 PRINT "3 -WAVES"
1080 PRINT "4 -ALERT"
1090 PRINT "5 -URGENT RUNNING"
1100 PRINT "6 -UFO MOVING"
```

```
1110 PRINT "7 -DEATH RAY"
1120 PRINT "8 -CRASH"
1130 PRINT "9 -BLAST OFF"
1140 PRINT "A -STRIKING CLOCK"
1150 PRINT "B -AMERICAN SIREN"
1160 PRINT "C -EUROPEAN SIREN"
1170 FOR I=1 TO 4
1180 PRINT
1190 NEXT I 1200 RETURN
1210 REM DELAY 1
1220 FOR DELAY=1 TO 200
1230 NEXT DELAY
1240 RETURN
```

Assignment Question 20-3

```
10 REM SONG
15 CALL CLEAR
20 FOR I=1 TO 27
22 READ P,D
24 CALL SOUND(D,P,10)
40 NEXT I
100 DATA 262,300,262,300,262,200,294,150,330,300
101 DATA 330,200,294,100,330,200,349,100,392,600
102 DATA 523,175,523,100,523,175,392,175,392,100
103 DATA 392,175,330,175,330,100,330,175,262,200
104 DATA 262,100,262,200,392,200,349,175,330,200
105 DATA 294,150,262,600
```

Bye for now.

`END OF ARTICLE`

PUT PEN TO PAPER & WRITE A TND ARTICLE

Diskname P097
Used=312 Free=46

This diskette contains Page Pro LINE FONTS. It also contains example pages of the line fonts.

BLOCK1-LN	3 Prog	BLOCK2-LN	3 Prog
BOLD-LN	3 Prog	D01-LN	3 Prog
D02-LN	3 Prog	DATA1-LN	3 Prog
DATA2-LN	3 Prog	F01-LN	3 Prog
GALAXY1-LN	3 Prog	GOTHIC1-LN	3 Prog
GOTHIC2-LN	3 Prog	HELP	21 Prog
ITALIC1-LN	3 Prog	ITALIC2-LN	3 Prog
LEDGER1-LN	3 Prog	LEDGER2-LN	3 Prog
LGF*LBL	12 I 13	OTLINE1-LN	3 Prog
OTLINE2-LN	3 Prog	PAGE01D	21 Prog
PAGE02D	21 Prog	PAGE03D	21 Prog
PAGE04D	21 Prog	PAGE05D	21 Prog
PAGE06D	21 Prog	PAGE07D	21 Prog
PAGE08D	21 Prog	PAGE09D	21 Prog
PAGE10D	21 Prog	PLAIN1-LN	3 Prog
PLAIN2-LN	3 Prog	SCRPT1-LN	3 Prog
SMCHARS-LN	3 Prog	T01-LN	3 Prog
TULO-LN	3 Prog		

DiskNAME P098
Used=350 Free=8

This diskette contains Page Pro BOARDERS. Boarders load as Pictures.

01C	18 I 13	01D	15 I 13
01E	10 I 13	01R	21 I 13
01S	15 I 13	01W	21 I 13
01X	10 I 13	01Z	18 I 13
02C	20 I 13	02D	11 I 13
02E	8 I 13	02R	20 I 13
02S	11 I 13	02W	20 I 13
02X	8 I 13	02Z	20 I 13
03C	20 I 13	03E	7 I 13
03R	20 I 13	03SD	10 I 13
03W	20 I 13	03X	7 I 13
03Z	20 I 13		

Diskname P099
Used=358 Free=0

This diskette contains Page Pro LARGE FONTS

3D1-LG	19 Prog	3D2-LG	19 Prog
BKMN1-LG	19 Prog	BKMN2-LG	19 Prog
BLOCK1-LG	19 Prog	BLOCK1-SM	6 Prog
BRDWY-LG	19 Prog	CONTOUR-LG	19 Prog
DATA1-LG	19 Prog	DATA1-SM	6 Prog
DATA2-LG	19 Prog	DATA2-SM	6 Prog

GALAXY1-LG	19 Prog	GALAXY1-SM	6 Prog
GOTHIC1-LG	19 Prog	GOTHIC1-SM	6 Prog
GOTHIC2-LG	19 Prog	GOTHIC2-SM	6 Prog
ITALIC1-LG	19 Prog	ITALIC1-SM	6 Prog
LEDGER1-LG	19 Prog	LEDGER1-SM	6 Prog
PLAIN1-LG	19 Prog	PLAIN1-SM	6 Prog
STNCL-LG	19 Prog		

Disk#2:G100
Used=303 Free=55

This diskette contains two games KARATE CHALLANGE. and MISSION DESTRUCT both are very good quailty games that now can be release due to Asgard stop supporting. TI.

CHARA1	9 Prog	DESTRUCT	33*Prog
DESTRUCU	33*Prog	DESTRUCV	31*Prog
GJS	11*Prog	KARATE	33*Prog
KARATF	33*Prog	KARATG	17*Prog
LOAD	5 Prog	LOAD/DES	35 Prog
LOAD/KAR	35*Prog	ROOT	28 Prog

END OF ARTICLE

EDITORS COMMENTS

I would like to thank Daniel for his letter and the information about the two different PC. clubs. Its good to know that our little computer (TI99/4A) and its members still hold their own against other computers.

For your interest in basic programming I am sure that there is somebody that can help with advice. This is what our club and magazine is about, what I would like to do is ask for anyone with problems, or who have had, and rectified a problem to please drop the magazine a line and share your experience. Maybe Daniel could tell us how he managed to overcome his particular difficulties.

Our last meeting was very well attended even though it was mid way through the holidays, Geoff Trott was pinned down with a soldering iron in one hand resistors in the other and solder from the mouth, Geoff was available for advice and help for anyone who could be having trouble fitting there 80 column cards. Peter had some IBM computers set up and in all stages of repair, in the adjacent room Ross had part of his LAN system set up, and Larry had his system up and running, demoonstrating his new printer to whoever had a spare nonent.

Dont forget that the november meeting is a BUY SWAP and SELL day so bring your gear and your money.

LETTER TO THE EDITOR

by Daniel Harris
(Edited by Bob Relyea)

My fellow members of TIsHUG,

Each month I receive the TND and a 'Printscreen' as I am a member of both TIsHUG and a PC Users Group. It is interesting to compare the approach of the two groups. With our group a lot of attention is given to projects but not enough Basic program writing to suit me. From my experience with printers and Basic program writing I have found it to be a bit of a tough road to hoe at times but I have persevered and have made various break-throughs along the way. You do have to have a bit of technical knowledge if you want to get into TI computing in a big way. TI seems to put a lot of emphasis on getting things to happen with a cartridge or a software compiler instead of getting it to happen from the Basic environment.

Compare this with the PC Users Group. They will do some programming now and then, but since their group contains a lot of computer professionals their publications contain articles which review the virtues of operating systems as well a lot about systems programming. Programs are not listed and an author charges a fee if his article gets a mention. I have not noticed anything for the soldering enthusiast, however! There exists a huge indexed catalogue of shareware, 24 Special Interest Groups (SIG), 3 BBS's, and the last time I went to a meeting there were 4,000 users present! As I understand it they are joining up at the rate of 1,000 each month, and swallowing up other users groups along the way. There might be a considerable amount of money to be made for projects if TIsHUG decided to become the TI-SIG within this group, for the same fee-$35 a year. It is like jumping from a fish tank into Sydney Harbour and can give you a lost child feeling! Very old TIsHUG members like myself can remember Shane Andersen and the Kings Cross days with a thousand member group with huge active SIG's. Well, that is nothing compared to what is on the scene now, and apparently, amalgamation is the way to go. Imagine a meeting where you could not meet all the members and you could run out of puff just chatting to 50 or so at the most? More crowded than the Royal Easter Show!

Daniel Harris
17 Thomas Street
HURSTVILLE, NSW, 2220

Techo-Time

from Geoff Trott

Creating Program Image Files

I have been buying the games disks from the shop (for my son, of course!) since Larry Saunders has been doing such a great job of packaging them up. For those who are not familiar, Larry puts on each disk a LOAD program which starts under Extended BASIC and then loads ROOT, which is a menu program that runs like the RAMdisk operating system MENU program. This then displays a menu of all the games on the disk which can be run by pressing the corresponding number. This works well for Extended BASIC programs and Editor Assembler option 5 programs but not so well for BASIC or Editor Assembler option 3 (Load and Run) programs, as these require extra steps to be taken. It should be possible (but not easy) to change the BASIC programs to Extended BASIC programs, depending on the character sets used, and it should be possible to convert the Load and Run programs to be memory image (option 5) programs. As two recent disks (G046 and G058) contained Load and Run programs, I decided to convert these to memory image or program files. I asked Larry about them and he explained that they were too hard to do in the time he had available, so this seemed to be a good thing for me to try and do.

The first thing I did was to examine these two programs. One of these was called Caverns and consisted of two files called CAVERNS1 and CAVERNS2. These files were Display Fixed 80 types and the first one was about 48 sectors long with the second one 93 sectors long. This program did not self start so a starting name had to be typed in. The loading process starts with the menu number which loads a file called ED/AS. This then shows the Editor Assembler menu with item 3 labelled CAVERNS. Pressing 3 brings up the prompt to enter the first file and the cursor has to be moved back to change the disk number from 5 to 1 and then type in the file name CAVERNS2. When this file has been read in, there is a prompt for another file, this time from DSK1. After typing in CAVERNS1 and waiting for it to read in, pressing enter at the next prompt causes the start name of CAVERN to appear and pressing enter again gets the program running. I have not been able to get the program running by putting the files in the other order unless I use Funnelweb or the Editor Assembler module.

The second of these programs is called Perfect Push. It is one Display Fixed 80 file of 91 sectors long. It starts the same way as Caverns up to typing in the first file name, except the file name is PUSH, and it starts automatically without any further typing. I first decided to make Caverns auto-start, which worked

OK but still required the two file names to be typed in so this is part of another story. I then decided to look at the files for these two programs and work out how to save them as program files. Saving programs as program files is done by using the SAVE program, which is a Display Fixed 80 file itself. SAVE requires that some symbols are defined before it is run. These are SFIRST, SLAST, SLOAD. SLOAD defines the first address of the program and SLAST the last address the program uses in memory. SFIRST must be the same as SLOAD and I assume was originally designed to be the starting address of the program but because of the constraints of the program file format, cannot be different from SLOAD. This means that the first instruction that will be executed is the first thing that is loaded into memory and so is the first data loaded into memory in the first file of the sequence of files that make up the memory image version of the program.

Memory image files can be more than one in number because they are restricted to 33 sectors (8 Kbytes) in size (normally, anyway), or because there are gaps in the memory used by the program (each file contains contiguous memory locations only). The names of the files that will be loaded as a continuous program, have names whose last character increases in its ASCII code by one, file by file. The SAVE program does this for us in the simplest cases. Funnelweb's FSAVE does it all a little better. FSAVE has several entry points: SAVE, which does the normal save function, but a little better; HBSAVE, which adjusts the high memory start to >A050 above the Mailbox; and MEMSAV, which allows the memory address limits to be entered, irrespective of the values in SFIRST and SLAST. There is good detail about FSAVE in the Funnelweb documentation on Utilities.

Back to the cases in point. When I examined the Caverns files using Funnelweb's DiskReview, I found that they were compressed format and used absolute addressing and data. To do this, I used the Inspect file function of DiskReview, looking at the offset in the file. This allowed each sector of the file to be looked at in hexadecimal or in ASCII. Since the file is in Display Fixed 80 format, each record is exactly 80 bytes (characters) long, which is >50. Since there are 16 items on each line of the display (whether in ASCII or hexadecimal format), each record takes up 5 lines and there are 3 records in each sector. Each record consists of a number of character tags followed by up to two fields. These character tags are ASCII characters while the data in the fields is in binary. The most used character tags are '9' for absolute addresses, 'B' for absolute data, 'A' for relocatable addresses, 'C' for relocatable data, and 'F' for the end of record. The first character tag in each record is an address tag and this is followed by a number of data tags indicating words of data (two bytes). Looking at a sector in ASCII mode and seeing lots of '9's and 'B's means that the

code is all absolute. To find the range of addresses used by a Display Fixed 80 file, go to the first sector and find the first '9'. Position the cursor on this '9' and change to hexadecimal mode. The address is then the 4 digits after the '39' on which the cursor is resident. To find the last address, go to the end of the file and look for the last '9' tag (in ASCII mode). This will be followed by a number of records with '3', '4', '5', and '6' tags, indicating External references and External definitions. Having found the last '9' tag, count the number of 'B' tags following it, determine the address following the '9' tag and add twice the number of 'B' tags following it (in hexadecimal, of course) to this address. The Editor Assembler manual explains more about these tags.

Using this approach, I found that CAVERNS1 contained data for addresses >D400 to >ED7A while CAVERNS2 contained data for addresses >A000 to >D3BA. CAVERNS1 had entries for ROCK, HAT and MINE while CAVERNS2 had an entry for CAVERN, the starting address at >A000. So I thought that this would be quite easy to convert to program format as it satisfied the requirement of the first data also being the first instruction to be executed. However it did have references (REFs) to routines like KSCAN, VDPWA, VDPWD, VDPRD which may require the presence of the Editor Assembler cartridge. That is something to be kept in mind. The first thing is to produce the program format file for the program itself. What is missing is the definition of the names, SLOAD, SFIRST, SLAST. From looking at the data with DiskReview, I know what values these must have so it is only a matter of making the loader aware of the values. To do this a small assembler language program needs to be written to define their values. This looks like the following.

```
      DEF   SFIRST,SLOAD,SLAST
SFIRST EQU  >A000
SLOAD EQU   >A000
SLAST EQU   >ED78
      RORG
      BSS   SLAST-SFIRST
      END
```

The first line says that the names listed are defined in this program. The next three lines define the values for the three names. The next two lines change the pointers used by the loader to point to the end of where the absolute code ends. This is so that when the save program is loaded, it will not load on top of the program we are trying to save. This is because the loader does not update the pointers for absolute code. If the code were relocatable (tags of 'A'), these two lines would not be needed and SLAST could be determined another way. This program needs to be typed in, assembled and then loaded. I called my program PFF;S for the source and PFF;O for the object file (DF80). The saving sequence was then: choose LOADERS option from the Funnelweb menu; choose Load and Run from

the next menu; type in DSK1.CAVERNS2, then DSK1.CAVERNS1, DSK1.PFF;O, DSK2.FSAVE (Funnelweb disk in DSK2), and then a blank line. Choose SAVE from the list of possible entry points, press CTRL[A] and then type in the file name you wish to use (DSK1.CAVERN). Three files are produced, CAVERN, CAVERO, and CAVERP. This can be tested to see if it needs the Editor Assembler cartridge by loading it using the Funnelweb loader in GPL mode. It worked fine like this so it obviously uses the functions available in Extended BASIC.

The second program was a bit trickier as it did not start at the first address (>A000) and it automatically started which does not allow SAVE to be loaded and run. Looking at the Display Fixed 80 file with DiskReview towards the end of the file, we need to find the record which has a tag character of '1'. This is the entry point and needs to have the tag character changed to 'F' to disable this feature. I found that the program used absolute addresses and memory from >A000 to >DCCE and had a starting address of >ADAA. Talking to Rolf about this, he said that his normal solution would be to note the contents of the first two words at >A000 and >A002 and replace them with a B @>DCCE. At the end of the program he would put in code which would restore the contents of the memory locations changed and then jump to the starting address. That is a clever way to do it but I thought of a simpler way by making use of the facilities of FSAVE. My idea was this: load all the files into memory as before (change PFF;S to suit and assemble); choose MEMSAV as the starting address and enter the addresses as from >ADAA to >DCCD and the file name as PUSHO. This will generate two files, PUSHO and PUSH1. After doing that, FSAVE returns to the loader and entry of a blank line followed by starting at MEMSAV again allows the rest of the program to be sent to PUSH2 from >A000 to >ADA9. Now the third file needs to be linked to the first two by using DiskReview to Inspect the first sector of the file PUSH1 and change the first word in the sector from >0000 to >FFFF. Now the result can be tested by loading via Funnelweb. In this case the program did not work in the GPL environment but did work in the Editor Assembler environment. So it is necessary to also load in the Editor Assembler code into low memory. Funnelweb has a file called EA which is just what is required, but it needs some doctoring.

First copy EA into the disk with the three other files and rename it as PUSH3. Inspect the file with DiskReview and change the second word to >07FA and the third word to >2000 being the size and address of the file. Then Inspect PUSH2 and change its first word from >0000 to >FFFF to allow the fourth file to be linked with the other three. Then all works well when loaded with any environment. It is a bit difficult to describe all the details in words, but if you have a small idea of what is going on, it should be easy to do this sort of thing with any program.

As a postscript, I also noticed that on one of the games disks (G032), there is another version of TI-Runner which starts with 30 men. This is supplied as two program files (RUN, RUO) and a Display Fixed 28 file (LEVEL28) of the different screens. However it is also run through the ED/AS program which made me think that it needs the Editor Assembler cartridge environment to run. So, I copied the PUSH3 file to this disk, renamed it RUP and changed the first word in the first sector of the RUO file from >0000 to >FFFF. Then the program loaded and ran without the need for the ED/AS file.

```
┌─────────────────────────────────────┐
│    END OF ARTICLE    ▨▨▨▨─
└─────────────────────────────────────┘
```

DID YOU KNOW ?

By Loren West

TELSTRA DISCOVERY.

What is Discovery?
Telstra Discovery is an electronic information service giving you access to a wide range of databases and interactive services. Discovery's flexiblity means that you can access a wealth of information within Australia and Overseas.
Discovery's service will open up a whole new communications world.
DISCOVERY 40
Discovery 40 is accessed through the phone number 01924 at 2400/2400 or 01955 at 1200/75 baud rate. It offers an easy to use format with 40 column text layout and extensive use of colour and graphics.
Discovery 40 information is arranged in a number of magazines which cover business information, travel, racing, leisure and computing.
MONEY WATCH
Money Watch is Discovery's financial servive with all the latest share prices, currency retes, futures, continually updated financial news and more.
BUSINESS RESOURCES
Business Resources contains many essential information resources including Electronic White and Yellow Pages directories, Commonwealth Bank Telebank service, Australia Bureau of Statistics, AAP news and a useful range of other database services.
RACING
Racing is a must for the serious punter who values the TAB's continually updated race results, online betting and form guides, as well as rating tipster services.

LEISURE and COMPUTING
Leisure and Computing contains a large range of services and informatoin related to leisure and computers including downloadable software, bulletin boards, online shopping and a market place for your software and hardware needs. For a bit of light relief, games, quizzes, bulletin boards and chat services, online shopping and more avaliable.

DISCOVERY 80
Discovery 80 is accessed through the phone number 01956 at 1200/1200 and 2400/2400 baud rate. Its ASCII format has 80 column text layout with no graphics or colour. Discovery 80 connects you to a large range of national and international database services for all your information needs including:

BUSINESS RESOURCES
Business Resources gives you access to Dow Jones News/Retrieval for the latest overseas financial information. Tenders information is available from the Australian Government Publishing Service, as well as the AGPS GOLD directory of Commonwealth Government personnel. Corporate information is also avaliable from the Australian Securities Commission's ASCOT database, including Business Names from NSW, WA, and QLD Registries.
PRONET
A large national and international bulletin board and information exchange service for an extensive range of special interest groups, particularly education and PC users. It also has extensive file and message uploading and downloading capabilities.
OAG
Access to the Official Airline Guides with the latest information on airline schedules, fares, accommodation and online bookings.
MESSAGING
Allows you to send electronic mail to other Discovery and Telecom Keylink users in Australia and via X400 protocol to overseas recipients, as well as sending and receiving telexes and sending faxes. Messaging also has uploading and downloading of text and binary file capabilities.
To find out more ring 008 028 339(free call) during business hours.
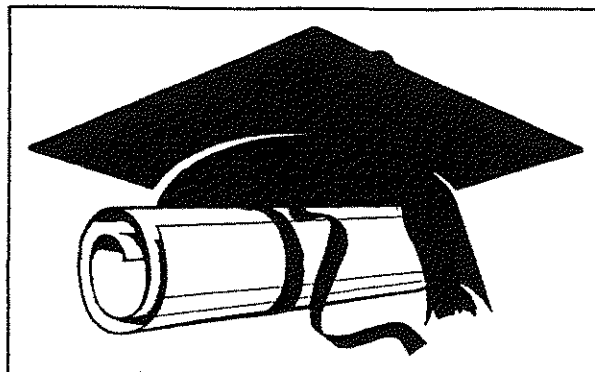
CORRECTIONS to SHAHZADA ENDURANCE HORSE RIDE ARTICLE.
by Ross Mudie, 27th October 1994.

In the October 1994 TIsHUG news digest, my article on
the computer system used at the Shahzada Endurance Ride
the data was corrupted between what was placed on the
BBS and the pages of the magazine. The paragraphs with
errors are reprinted herewith.

Page 11, left column, paragraph 2.

```
            SERVER   CABLE   REMOTE TERMINAL

Protective Earth    1 ----->)<--- 1  Protective Earth
        Rx Data     2 (14)--<--- 3  Tx Data
        Tx Data     3 (16)-->--- 2  Rx Data
    Signal Earth    7 ----->)<--- 7  Signal Earth
Data  Carrier  Detect 8 (12)-->---20  Data   Terminal
Ready
Data  Terminal  Ready 20 (19)--<--- 8  Data   Carrier
Detect
```

Page 11, right column, paragraph 3

When a batch application such as PRINT, BACKUP or SORT
is invoked, an appropriate flag is set in the server so
that the function can be treated in the same way as an
additional   calling   terminal.   All   three   batch
applications are broken up into one record per scan to
allow other terminals or batch applications to be
handled without having to use interrupts in the
computer program. Once any batch application has been
commenced by a terminal, the terminal is free to be
used for other terminal functions which may include
data entry, starting another batch application or
cancelling an already running batch application.

Page 12, left column, paragraph 3

When a terminal was not plugged in to the serial port
of the server or a terminal program was interrupted in
operation by a FCTN clear, it was possible for the DTR
line on that terminal input to the server to remain in
a positive voltage state which was treated as a
permanent call condition. This in turn would stop the
scan program which stopped the operation of the LAN.
To overcome this problem, the scan program will only
wait on a 'stuck high' input for 0.5 seconds and then
resume the scan. If an input is found to be 'stuck
high' for 6 successive scans, it is 'locked out' until
it resets to low. In the locked out state an input is
still tested once every scan, but the scan program does
not wait for the 0.5 second period until it is reset to
low for one scan. The self resetting nature of the

software means that as soon as a terminal returns to
normal, the server will start to handle it. The
terminal lockout counter can also be reset from the
server keyboard.

page 12, right column, paragraph 5.

An essential tool in setting up for computers in a
horse float is a shovel to remove the horse droppings
first! After the environmental conditions have been
reduced to just dusty and not too squelchy under foot,
its time to bring in the computer table, chair and the
computer. All of the cables to the server from the
terminals were just wound off the drum. This required
that the 25 pin connectors were terminated on site with
soldering iron and pliers once the cables were in
place.

Page 13, left column, paragraph 4 and 5, which merged
together.

A stand alone RS232, (TI boxcar unit), suffered from
moisture ingress which caused corrosive growth between
tracks of the lower PCB at a previous Shahzada. The
problem was found and scraped clean prior to this
Shahzada. Even though the computer was located in a
tarped up horse float and covered with a tarp at night,
there was still enough moisture in the air to cause a
re-growth of the corrosive problem between the tracks
and it failed in full sunlight conditions. That PC
board has again been cleaned and coated with a
protective lacquer to try to prevent a re-occurance of
the problem (after the event). To overcome this
problem quickly in the event, I did a quick
re-configuration of equipment utilising spares which I
had available on site and quickly made a 'Y' cable
adaptor for the RS232. Unfortunately this change did
not allow continued use of the local logging printer at
the time keeper because of the lack of a parallel cable
or the parts to make one on the spot. The people using
the Time Keeper's terminal were sufficiently familiar
with the system to get by nicely without this printer
at this stage of the event. After this the terminals
in the horse floats were left powered up over night to
maintain some warmth in the computers and prevent
further problems with condensation.

The secretary's logging printer also failed to produce
the goods twice. The first was when the printer was
moved by people from the Secretary's area and the
serial cord was dislodged. The other case was a coat
placed on top of the printer which jammed the paper.

END OF ARTICLE

As-Easy-As ...

Options

The Options submenu provides a variety of choices. Here, for example, you can specify a legend for a data range (for example, what the individual bars on a graph represent) with the Legends option. The Titles option allows you to put a label on both sides and give the entire graph a title.

Legends help identify what the individual data ranges are. They are placed under the graph and appear, in our example, with a small box containing the same cross-hatch pattern as the bar.

The Format submenu gives you more choices on the type of display. The values can be indicated by lines, shown as symbols for individual values, or as a result of a combination of lines and symbols.

The Titles submenu enables you to place graph titles above, below, or on the side of the graph.

The Grid submenu lets you set up a grid of your graph coordinates. The grid display is drawn with lines.

The Scale option establishes the measurements for mapping out the X and Y axes. The Y axis is scaled so that the largest value can be presented. Besides automatic scaling, you can also manually pick the lower and upper limits of indicated values. Choose the Manual operation, then Upper and Lower for the X or Y axis. The Reset operation from the main graph manual lets you reset individual graph parameters.

Name submenu

With this menu item you can name and save your graph (CREATE), activate (USE) an already existing graph, or erase a graph (DELETE). You can also put together or split apart two or more graphs by using the MERGE, SPLIT, and RESET options.

View and Plot commands

The View option displays your graph on the screen, while the Plot option sends your graph to the printer.

You can also display your graph on the screen without calling up any menus by entering <F10>.

4.4.2 Other graph options

```
F1:Help 2:Edit 3:Macro 4:Abs -Graph- 5:Goto 6:Window 7:Wp 8:Calc F10:Graph
X-Y,Bar,Pie,Stack,Line,Cum,Htoc,Radar,Area,Delta
/.......B/.......C/.......D/.......E/.......F/.......G/.......
GRAPHICS
        t   14.50%
        $6,000.00
Type
X          36 months
A          $206.53 monthly
B
C        Interest Principal Remaining    Total     Total
D          Paid     Paid    Principal   Interest Principal
E         $72.50  $134.03  $5,865.97    $72.50   $134.03
F         $70.88  $134.03  $5,731.85   $143.38   $268.05
Labels    $69.26  $135.65  $5,596.30   $212.64   $403.70
Options   $67.62  $137.26  $5,459.04   $280.26   $540.96
Reset     $65.96  $138.90  $5,320.13   $346.23   $679.87
Name      $64.28  $140.56  $5,179.57   $410.51   $820.43
View      $62.59  $142.24  $5,037.33   $473.10   $962.67
Plot      $60.87  $143.94  $4,893.39   $533.97 $1,106.61
Quit      $59.13  $145.66  $4,747.73   $593.08 $1,252.27
          $57.37  $147.40  $4,600.34   $650.46 $1,399.66
19  11    $55.59  $149.16  $4,451.18   $706.05 $1,548.82
20  12    $53.78  $150.94  $4,300.24   $758.84 $1,689.76
Free: 98% [261k]   Auto PD800K WKS        Num       10:26:12 am
```

Graphics Main Menu

The AsEasyAs Graphics program provides several options. These options are available in submenus.

Graph Type

In this submenu you decide the graph type, such as bar, pie, line, or X-Y plot. However, only one kind of graph can be created at a time.
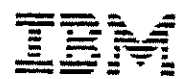
X-axis

This sets up the graph range for the X-axis. The letters X, A, B, C, D, E and F represent the various ranges which can be represented as bars in a graph. The graph range refers to a row of related data. In our example, we choose the range of cells from A9 to A20. The labels are then assigned to the X range. They can be either values or text and are placed under the horizontal axis.

A range

Initializes the first graph range for the vertical or Y-axis (in our example, cells C9 to C20). The B through F ranges allow you to add a second through sixth vertical range.

4.5 Working With Macros

With macros, almost anyone can use AS-EASY-AS even if he/she doesn't completely understand the program. Users can automatically execute entire series of program operations by defining macros. However, the method of macro programming is difficult to learn. You should have some knowledge of at least simple programming before proceeding in this section.

This process of collecting a series of commands into one name is called creating a macro. A macro is designed to help automate the program.

What is a keyboard macro?

Keyboard macros refer to a sequence of keystrokes that are stored with the worksheet and perform certain repeating operations. These sequences of keystrokes have a name which enables you to call up these commands by typing <Alt> and the appropriate key.

For example, <Alt><A> starts a macro named "A".

A macro with the name "\0" will automatically be executed when AS-EASY-AS begins.

This macro can now be called by entering <Alt><S> or <F3> and the cell name on which it was entered.

Macro Keys

You can use the following options within the command:

[LT]	Cursor left
[RT]	Cursor right
[UP]	Cursor up
[DN]	Cursor down
[PGLT]	Scroll one screen left
[PGRT]	Scroll one screen right
[PGUP]	Scroll one screen up
[PGDN]	Scroll one screen down
[EDIT]	Turn on <F2>
[ABS]	Turn on <F4>
[CALC]	Turn on <F9>
[NAME]	Insert file name
[GOTO]	Turn on F5
[GRAPH]	Turn on <F10>
[ESC]	Turn on <Esc> key
[WINDOW]	Turn on <F6>
~	Same as <Enter>
[?]	Enter data

4.5.1 Recording and starting macros

For example, let's create a macro to save the worksheet. Move the cursor to an empty and unprotected cell. This is where you want to place your macro. Next press </> to access the main menu.

Enter:

WSheet	Worksheet command
Macro	Macro command
Compose	Start writing a new macro command
Type (/F~S)	<Enter>
<Ctrl><M>	Leave the macro mode
/RN	Range name

4.6 Database Management

You can place data in a database by using the "Data" option from the main menu. A database in AsEasyAs is a table which is sorted or interpreted according to certain criteria. In the worksheet, each data record contains a line with a specific structure.

Organizing data isn't one of AS-EASY-AS' best features. For this kind of application you should use a database management program. PC-File+ (refer to Chapter 3) is better suited for this purpose. For this reason, we will only give you basic information on how to do a data sort in AS-EASY-AS.

First you must define the range to be sorted with "D-Range" (under "Data" and "Sort"). Next, select one or two sort fields using the "P(1)-key" and "S(2)-key" commands, determine whether the items should be sorted in ascending or descending order, and then execute the sort procedure with the "Go" command. You can also use these procedures when you work with normal tables.

```
F:Help 2:Edit 3:Macro 4:Abs Rel#r  5:Goto 6:Window 7:Wp 8:Calc F10:Graph
A5 [W15]: 'Sanders
Indicate PRIME Key: A3..A5    Sort Order (A) or (D):A
[ A ]..Al.......Bl........Cl.......D/ El.....Fl........
  LAST NAME   FIRST NAME   ADDRESS         CITY       ST  ZIP
1
2
3Fielder    Cecil    Tiger Stadium    Detroit       MI 48005
4Sanders    Barry    Silverdome       Pontiac       MI 48097
5Thomas     Isiah    The Palace       Auburn Hills  MI 48234
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
Free:100% (265k)  Auto                     Num :          10:50:28 am
```

Sorting a database

4.7 Printing

Select the "PrintTo" option from the main menu for a hardcopy of your spreadsheet. Then select whether to send your spreadsheet directly to the printer or send it as part of a new or existing text file (for example, if you want to use it in a report you already created with PC-Write).

Before you start printing with the "Go" command, you should first select the range [Wide 0: Pages 0]. You can also change the print layout. With the "Options" function, you can set margins, define headers or footers, and select the page length. Select "Setup" to send a printer code to the printer (e.g., a code to change the print style).

4.8 Importing Data

AsEasyAs allows you to import data from other databases. To do this, enter "File" and "Import" from the main menu.

If you select the "DbIII" option (dBASE III), records in your selected dBASE format file will be added to your new worksheet. Each field of the records in the dBASE file will be placed in a column of the worksheet, with the column widths automatically adjusted for the field size.

If you select the "Values" or "Text" option, AsEasyAs assumes that you're attempting to import an ASCII file. Under "Text" each line of the imported file becomes a string in the current column of the worksheet. These strings can then be broken down by using the "Parse" command. When you select the "Value" option each individual field from the imported file is placed in a different cell on the spreadsheet.

Note: Importing occurs at the cursor location.

REGIONAL GROUP REPORTS

Meeting Summary For NOVEMBER

Central Coast	12/11/94 Saratoga
Glebe	10/11/94 Glebe
Hunter Valley	13/11 20/11/94
Illawarra	15/11/94 Keiraville
Liverpool	11/11/94 Yagoona West
Sutherland	18/11/94 Jannali

==

CENTRAL COAST Regional Group

Regular meetings are normally held on the second Saturday of each month, 6.30pm at the home of John Coulton, 34 Mimosa Ave., Saratoga, (043) 69 3990. Contact Russell Welham (043)92 4000.

==

GLEBE Regional Group

Regular meetings are normally on the Thursday evening following the first Saturday of the month, at 8pm at 43 Boyce Street, Glebe. Contact Mike Slattery, (02) 692 8162.

==

HUNTER VALLEY Regional Group

The Meetings are usually held on the second or third Sunday of each month at members homes starting at 3pm. Check the location with Geoff Phillips by leaving a message on (049) 428 617. Please note that the previous phone number (049) 428 176 is now used exclusively by the ZZAP BBS which also has TI support. Geoff.

==

ILLAWARRA Regional Group

Regular meetings are normally held on the second Tuesday of each month after the TIsHUG Sydney meeting at 7.30pm, at the home of Geoff & Heather Trott, 20 Robsons Road, Keiraville. A variety of activities accompany our meetings, including Word Processing, Spreadsheets and hardware repairs. Contact Geoff Trott on (042) 29 6629 for more information.

==

*   LIVERPOOL Regional Group   *

Regular meeting date is the Friday folling the TIshug Sydney meeting at 7.30 pm. Contact Larry Saunders (02) 644-7377 (home). After 9.30 PM or at work (02)602 3312 Liquorland Liverpool West for more information.

*** ALL WELCOME ***

11th November 1994
My Place : 34 Colechin St. Yagoona West

9th December 1994
My Place : 34 Colechin St. Yagoona West

Bye for now Larry.
Liverpool Regional Co-Ordinator

==

SUTHERLAND Regional Group

Regular meetings are held on the third Friday of each month at the home of Peter Young, 51 Jannali Avenue, Jannali at 7.30pm. Peter Young.

==

TIsHUG in Sydney

Monthly meetings start promptly at 2pm (except for full day tutorials) on the first Saturday of the month that is not part of a long weekend. They are held at the MEADOWBANK PRIMARY SCHOOL, on the corner of Thistle Street and Belmore Street, Meadowbank. Cars can enter from Gale Street and park in the school grounds. Regular items include news from the directors, the publications library, the shop, and demonstrations of monthly software.

NOVEMBER MEETING - 5th NOVEMBER

DECEMBER MEETING - 3rd DECEMBER

****

The cut-off dates for submitting articles to the Editor for the TND via the BBS or otherwise are:

December	-	12th November
February	-	14th January

These dates are all Saturdays and there is no guarantee that they will make the magazine unless they are uploaded by 6:00 pm, at the latest. Longer articles should be to hand well before the above dates to ensure there is time to edit them.

****
****
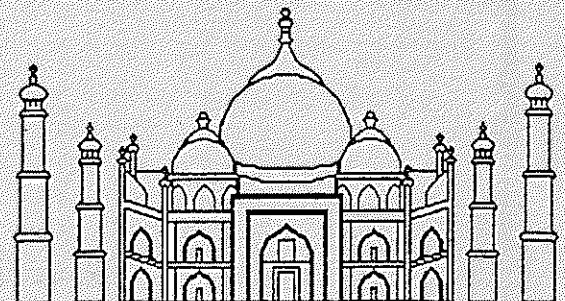
JUST A ONELINER.
(ED)

One fine day in the middle of the night.
Two dead men got up to fight.
Back to back they faced each other, Drew their swords and shot each other.