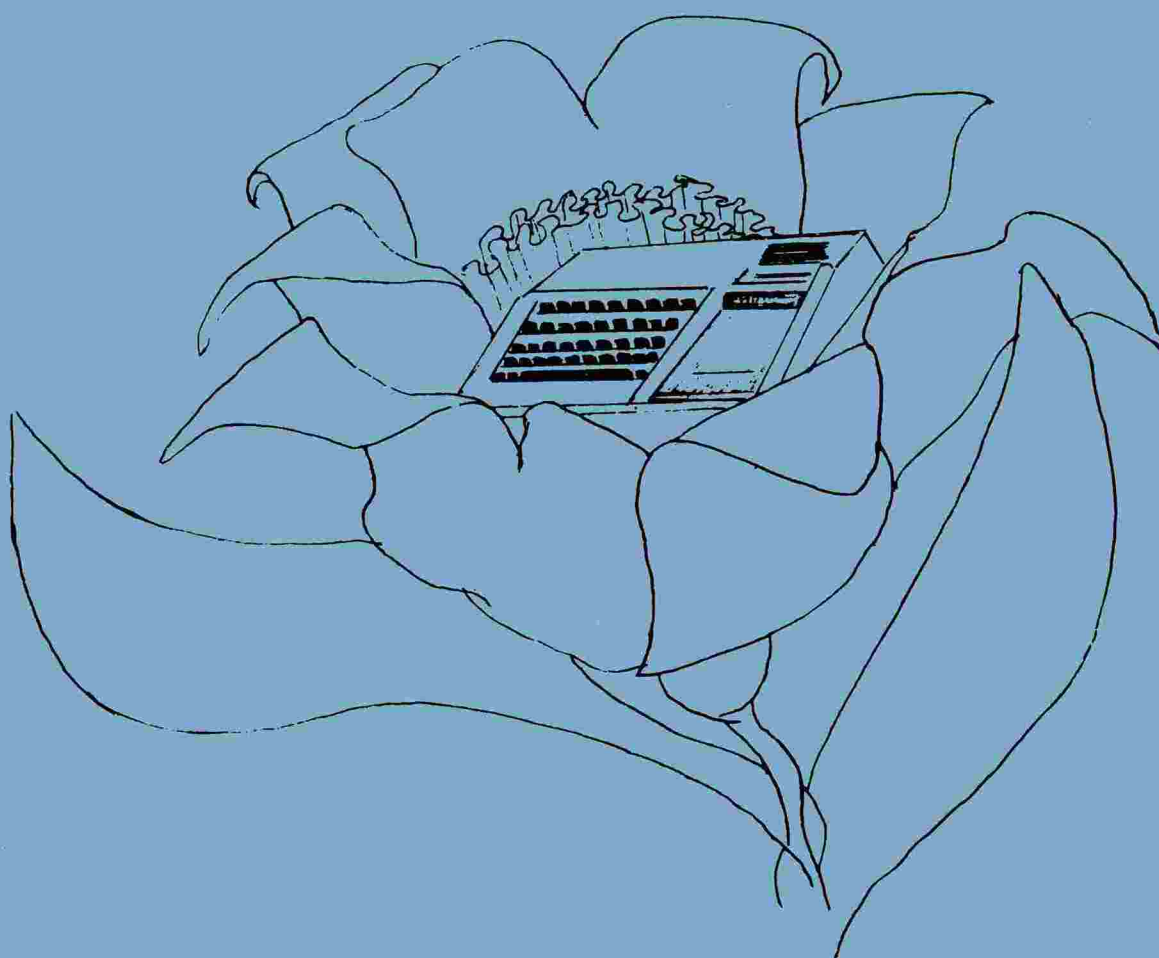


TI * MIES



A Future Flowering
NUMBER 17 SUMMER 1987

IN THIS ISSUE.....

PAGE

1	Editorial
5	Introducing the new committee
8	Chairmans TI99ENS Worth
9	Rambles
19	Tips from the Tigercub
21	TI Writer Tony McGovern
23	Program BASIC R Marshall
24	Life at Funnelweb Farm Mrs McGovern
29	Cassette Library Update
34	Bugs in R. Molesworth's Book
35	Program demo Sprites
36	Diagnose Your TI MSP 99ers
37	Faster cassette files Hunter Valley
38	Program X BASIC Flintstones
40	A Look at Memory Ottawa Users
42	Basic Beginnings-Variables
44	TI Writer MSP 99ers Minnesota
47	Program X BASIC TI-Timer
48	DIY Hardware
49	Program BASIC, X BASIC Spider Bop
52	TI Writer Notes TI Users Perth
55	Multiplan Ottawa Users
58	Program X BASIC+32k+Disk

TI99/4a USERS GROUP UK and TI*MES magazine is supported only by its subscribers. This TI users Group is INDEPENDANT of Texas Instruments and is completely non profit making. TI*MES is published quarterly. The annual subscription runs May to May each year. Details from the Group secretary on application. Editorial etc is provided by group members, other WORLD WIDE TI User Groups and other related sources. Views expressed are those of the writer and not necessarily those of TI994a Users Group UK. Whilst efforts are made to ensure accuracy no responsibility can be accepted by TI994a Users Group UK as a result of applying such information found within the pages of TI*MES. You are invited to contribute copy for publication in TI*MES. Please submit copy on A4 ONLY. This must be TYPED AND SUPPORTED WITH A DISK OR TAPE if a program is included. Last date of acceptance of copy for next issue is 30TH AUGUST FOR PUBLICATION IN OCTOBER. Unaccepted material will be returned ONLY if accompanied by a S.A.E. No material may be reproduced without credit to the author and TI994a USERS GROUP UK. MICRO COMPUTERS OTHER THAN TI99 COMPATIBLES WILL NOT BE ADVERTISED FOR SALE IN TI*MES.

4, Thornton Lane, Ulceby, South Humberside DN39 6SR(046 98 404)

T I 9 9 / 4 a U S E R S G R O U P U . K .

A Future Flowering

EDITORIAL

The TI Users Faire at Derby, 16th May, 1987

The Central Suite was easy to find, and I was grateful for that after a two and a half hour train journey. The Hall was spacious with tables set out around each side for the use of the dealers and for demo's. Rows of chairs were ready for The Meeting, which would see the hand-over of TI99/4a Exchange to become TI99/4a Users Group UK, and the formal adoption of a new committee to run it. It was only just ten'o'clock, but everyone was ready for business.

So, with a child on each arm, I took the two games modules I'd got when I bought our TI, to the Jumble Sale at the TI Exchange counter, this being patiently and efficiently run by Audrey Scally and daughter. Hopefully, these would be sold and we'd be able to buy some new ones. There were lots of software bargains, new and secondhand, and we managed to snap up a speech synthesizer and speech editor for less than half the normal price.

The display on show by Parco Electrics was most impressive and the children made a bee-line for the second-hand games which you were able to try out on the computer before you bought them. There was a good range of books, software and some hardware, and advice freely given if needed. A natty little gadget caught my eye - a language translator - with the same neat, slim design that we are used to from Texas Instruments, but small enough to pop into your handbag (or brief case!). It takes tiny plug-in modules of French, German or Spanish, and at a bargain price, with earphones, mains adaptor and free module who could resist?!

At the next counter was the classy-looking Geneve computer, kindly brought by Richard Sierakowski, many thanks to him, and some copies of TI-LINES to look at, also a chance to chat to the editor of that excellent publication, Peter Brooks, if you didn't mind joining the queue!

TI Exchange had the full expansion system for demonstrations, complete with printer and modem, none of which I had ever seen before. It a looked very neat and technical. Alas, no-one was selling any cheap expansion, so I'm still only sweet sixteen K! All through the day there was a sizeable crowd of supporters filtering backwards and forwards. A lively atmosphere was created by their enthusiasm and energy, characteristics shared by TI'ers everywhere.



At eleven o'clock The Meeting began, chaired by Clive Scally, founder of UK TI Users and TI Exchange. We were reminded of Peter Brook's offer (as printed in last TI*MES) of taking on the membership of TI Exchange, so that it's cohesive structure, built up under four years of superb effort by Clive and Audrey, would not be interrupted, and that it be incorporated into the International TI Users Group. All the benefits of that group, which is a "commercial" user group, would be enjoyed by TI*MES readers, who could contribute to TI-LINES. If they still wanted a "club" as such, people could form their own local users groups, and representatives from each group could write for TI-LINES.

Lively debate followed. Doubts were raised such as: would we lose the friendliness and exchange of opinions and ideas that we had as a national users club? We would lose the "blanket" of our own magazine, and the momentum of the larger group might be lost as it refracted into small local groups, if these were ever formed.

It takes a lot of energy to run a group and many people don't have the time, even to write a letter. We would lose our annual Faire, that golden opportunity to meet and actively show our interest and come together as a club (and in my case to trade new games for old!) Another worry was the financial survival of a commercial group, as against that of the club funded by it's members subscriptions.

Helpful information from Peter Brooks himself stimulated yet more debate!! It was discovered that quite a few people already subscribed to both magazines, therefore a need must exist for both. Clive pointed out that he had already received one hundred and sixteen new subscriptions for TI*MES, and it became clear that the commitment was still there. The new group found a focus for it's energies when Clive consented to become Chairman for the first year. The new committee was formally adopted and TI Users Group UK was born!

After lunch with my family at Uncle Mac's across the road (and a detour via the toy shop), the committee got together. And a lovely lot of chaps they are too! You're in strong, safe hands, dear readers! A formal constitution was hammered out, with nice things such as "pressure group to customer services at Texas Instruments" and "future development of modem use and bulletin boards" being mentioned. A copy is included for your comments in next issue.



The next event was an exciting auction of TI Exchange left-over stock, all proceeds to fund the new Users Group. Games were going for 50p, 25p and a console with a slightly damaged key went for one pound! We spent our last few pennies and got ready to go. I had come as an interested supporter and going home as an Editor with new responsibilities (and a lighter purse!) Despite great efforts the link-up with USA and Canada wasn't possible. although

contact had been made. What a full, exciting day we'd all had! Many thanks to the supporters and dealers who made it so, and special thanks to the Secretary for finding the venue. And what can I say to our founder Clive Scally for getting us all this far? We wouldn't be here were it not for your efforts and commitment, so, on behalf of everyone, committee and members, please accept our gratitude, Clive and Audrey.



Thanks ...
and
Greetings!

Now, the important stuff over, I'll introduce myself. I'm your new editor, Christina Mehew. I am married with two children aged nine and eleven. We live out in the wilds on the East Coast. We have owned a TI99/4a since 1982 which the children have used more than me, mainly because having worked with computers I couldn't face them when I got home! Also, being used to the full equipment at my fingertips - disk drive and printers - it annoyed me not having the 'gear' for my own. I've stuck with it because I know through reading our mag and through odd encounters with other micros, that the TI has such potential, even in its unexpanded state. It is so fast to load from cassette and has no nonsense about it. Others I've found slow to load and gimmicky in their own ways. They have to be marketed, I suppose.

So now I'm not working I have the chance to get to know Tex a little better! I'll knuckle down and learn TI Basic, combine it with my hobby of writing, and try to pass on this knowledge to those of you with newly acquired machines. Yes, folks, you've now got someone on your side! I've got some very valuable help too: co-editor Alan Bailey, and together we will try to give you a useful, informative magazine. Stephen Shaw will edit his "Rambles" as usual, and we would love contributions from YOU!

If a know-nothing like me can put key to paper, so can you! I won't apologise for not being a TI expert, as we've got some really good ones (still room for more, I'm not giving you an excuse not to write!) And I won't apologise for the flowers either! I've just seen some floppy disks marketed as bananas, of all things, with a free banana pen or Mercedes truck for the appropriate quantity! If they can have bananas, we can have our flowers. Besides, you men, when you lift you eyes up in boredom at the next set of flowers, its a cue for your wives to ask you to mend the washer! I'm on their side too!

Back to letters - USE THIS SPACE! Articles or just letters appreciated. Ideas I have that we could work on together: hints and tips, childrens page, general interest in the computer world. The only constraint is space, it's your user group. As James Baldwin, the American psychologist, once said: "The world is before you, and you need not take it or leave it as it was when you came in". Here it is, the Future Flowering - make it what you want it to be.

Christina Mehew, editor

INTRODUCING YOUR NEW COMMITTEE :

General Secretary - Jim Ballinger,
5, Offerton Avenue, Derby DE3 8DU
Jim is now retired from Rolls
Royce Ltd., and is a radio ham.
He has a full TI system and printer.



Joint Editor - Alan Bailey, 14,
Shelley Grove, Loughton, Essex.

Alan is a process engineer, now nearing retirement. He owns an unexpanded TI, and is interested in hardware and construction, computer languages, graphics and CAD.

International Journals, "Rambles", Disc Library - Stephen Shaw,
10, Alstone Road, Stockport, Cheshire SK4 5AH. Stephen was one of the first purchasers of the PAL TI99/4 and a founder member of the first UK User Group, TI-HOME. He is the proprietor of Stainless Software, and author of a TI book. Family man and a bank clerk, Stephen has a huge disk collection shared through the disk library. Latest purchases include Myark X3 & 512kram card.

GETTING ACQUAINTED

Chairman - Clive Scally, 40, Barrhill, Patcham, Brighton, E.Sussex BN1 8UF. Founder of TI*MES and TI Exchange Clive continues to give us his valuable support. Owns full system and modem.

Cassette Librarian - Maurice Rymill, 231, Bournville lane, Bournville, Birmingham B30 1RA. Maurice is a family man and works 12 hour shifts, but has found time to type and de-bug all the programs that were published in magazines in days gone by, and now has over 100 cassettes, in addition to the TI*MES library. Maurice is going to review some of these in TI*MES. He has recently expanded his TI, and has built the 32k matchbox expansion into his original keyboard.

Membership Secretary - Peter Walker, 24, Bacons Drive, Cuffley, Herts EN6 4DU. Peter started computing 21 years ago, using FORTRAN on a Honeywell 120 mainframe. He has full expansion, printer and modem for his TI, and has written lots of "practical" programs, such as accounts, and uses TI Writer, Multiplan and various database programs often. Peter is always happy to help with using data files.

Communications - Neville Bosworth, 61, Porchester Road, Woolston, Southampton, Hampshire SO2 7JB. Neville has full system, and now modem, since before the TI was orphaned, and has had not one moments regret in owning a system with such a superb design. He believes that the future of the TI and of all other computers is in the field of modem communications where enormous quantities of information can be exchanged between users. Will help anyone wishing to go 'on line', but interested in all aspects of the TI, as he believes that no knowledge is wasted.

Programming Languages - Geoffrey Coan, 66, Sanderling Close, Letchworth, Herts SG6 4HZ. Geoffrey is working with computers at ICL as this is his Industrial year, as part of his studies in maths and science at York University. He has expansion and disc drive and offers his help with programming.

Treasurer - Peter Cross, 8, Broom Mead, Bexleyheath, Kent DA6 7NY
Our very able treasurer has already sorted out our bank account and is holding on to the purse strings.

GETTING ACQUAINTED

Hardware and DIY projects - Mike Goddard, 2, Cysgod-y-Coleg, Bala, Gwnedd LL23 7BD. Mike has already given us some home projects, and hopes to develop this to supplying kits for these.

Committee Member - Edward Shaw, Crow Holt Farm, Basford, Leek, Staffs ST13 7DU. Edward is Quality Manager at a large mineral processing company and uses their Apricot X10. He has a Shinwa CPA 80 printer and a Pace Nightingale modem, and is a big fan of Art Green's Macro Assembler.

Committee Member - Tim Anderson, Flat 20/1, 33, Petershill Drive, Glasgow G21 4GH. Tim is a Medical Photographer, Bsc.(hons) in Photographic Sciences, but looking for something different! He now has a second disk drive for his TI, and 128k Myarc card. Tim has had a program published in Parco magazine and his idea is to create a system similar to that on the Apple Macintosh, but is limited by lack of knowledge of assembly language, which he is trying to learn, but can't seem to 'stick with it'!

Committee Member - Bryan Cloud, 27, Whitehouse Road, Ipswich, Suffolk IP1 5LS. Bryan works as a Field Service Engineer, mostly on X-ray machines, and has TI and expansion, two disk drives.

Here is a very able committee, willing to share their knowledge, (SAE, please, if writing to anyone), and they hope you will do the same!

ABSORBED?!!
IN
THE
OPEN
MEETING
AT
DERBY,
16TH MAY



CHAIRMAN'S TI99ENS WORTH

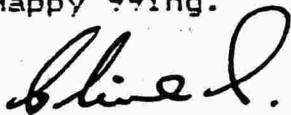
It is with great pleasure that I welcome all who have joined our happy 99er Group for yet another year.

Also it is a pleasure to thank most sincerely those on the new committee who have worked so very hard in administration and production of your fresh issue of TI*MES.

This issue has been produced by an Editor who really cares deeply about the TI99/4a, remarkably without use of an expansion system which I personally commend. (Can Christina become the Regina of U.K.?).

Now there is a full committee, everyone will benefit. It is over to you the TI Users to ensure the future of this Nationwide TI Users GROUP with involvement and support, we have that and thank you, just keep up the communication so vital to its existence where ever you are.

Happy 99ing.



Clive Scally.
Chairman, TI99/4a Users Group U.K.

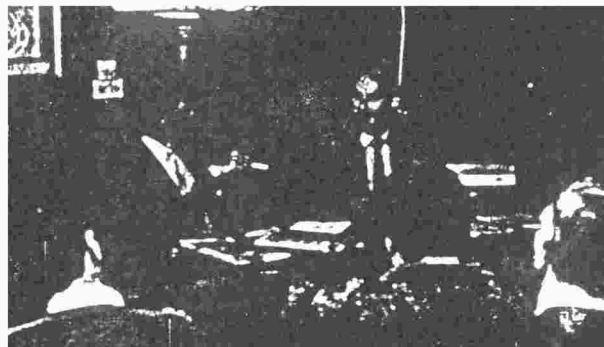
PS Did you see Chip Chat in July issue of PERSONAL COMPUTER WORLD? Your TI99/4a is pictured even today!

* * *

Pictured below: Clive Scally and Peter Brooks at the Open Meeting, Derby 16th May.

COMPETITION: What is Clive really saying to Peter?!!

A TI Module is offered as a prize for the wittiest answer, so get your pens out!



TI*MES

Rambles SUMMER 1987

by Stephen Shaw

Hello and welcome to another Rambles, perhaps a little late but there has been some organization and consolidation going on! As a result YOUR user group should now be in a more secure position than ever before!

The NEW group exists to serve its members, and is composed OF its members! If there is anything you want (but aren't getting!) SAY SO! And remember — EVERY member has SOMETHING to contribute. If you can't make a regular column, write in and say what you are doing, what your successes and failures are, what you use your TI for ...SHARE.

And speaking of sharing ... this last quarter (well ... quarter plus) has been quite unrewarding in terms of programs sent to me by UK users. Remember the long list I gave in the last TI*MES! This time the list is this long. Hamm. Not so long eh!

In obtaining new programs for YOUR user library, because we do not have UK-written programs to share with the world TI community, valuable pounds are being expended! AND UK users are getting a reputation as takers instead of sharers. How about it? If there is anyone out there, writing programs in ANYnguage for ANY purpose, SEND THEM IN.

My address is:

10 Alstone Road, Stockport, Cheshire, SK4 5AH.

CONTEST

And to give you a little inspiration, I am going to offer a PRIZE! (Maybe more than one depending on response, A box of ten double-sided disks goes to the sender of the most useful OR entertaining program in any language sent to me before November 10. No disk drive? OK, for unexpanded or younger owners, I have some Basic cassettes. This is open to everyone. The more entries, the higher the standard, then the more prizes. And if I only have one entry, or if the standard is TOO low, I reserve the right to carry the prize forward! This is for EVERYONE, from 6 to 60, AND both sides of that!

If you have to send cassettes, it is helpful if you can record your program twice, using a DIFFERENT tape recorder to record the second recording!

The rules:

All entrants must be paid up members of TI*MES.

Entries may be on DISK (SS or DS but must be single density) or cassette (see above).

Due regard will be taken of language used, age of entrant if given, originality, friendliness and usefulness or entertainment, Graphics and fonts and Multiplan templates and music and ... everything is welcome!

Contributions of unwanted nodules, hardware, etc, as extra prizes are also welcome! The more we can offer the better the quality.

I have suggested setting up libraries of modules and books/magazines for our members. If you are interested in such an idea, write in and say so. If you have books, magazines, or modules to contribute write in and say so. Some form of deposit seems a good idea to cover possible loss, and charges would be made to at least cover costs, and preferably make a contribution towards club funds. This little publication is expensive to produce!

How about a membership drive? The more members we have. sharing and contributing, the better we can serve each other. Do you know another TI user? Is he or she a member? Why not? What can we do to make it worth their while? No, we can't print money ...

No doubt much of this is also being written in the other sections of the newsletter. It can bear repetition!

Derby was a good get-together, but I was very surprised at the lack of goods for auction or sale. Obviously our members do not wish to part with anything! If there could be some item you no longer need, any not pass it on to another member!

I am still waiting to see a modem actually working! The frequent attempts to use a modem at our get-togethers has so far been fruitless! If there are any members using a modem, and happy with the result! please write in and tell us!

MICROpendium.

Have you used the MICROpendium subscription coupon in the last issue of TI*MES?

It is a useful magazine for keeping up to date with new software and hardware, and for you unexpanded folk there is our old friend Regena! MICROpendium has the best publication schedule of any TI-specific offerings, commercial or user group.

The Smart Programmer, now coming from Bytemaster, seems to have gone astray somewhere. The owner tells me he has work problems (you didn't think he was doing it full time did you?). We shall see what transpires.

Software Library

Many more offerings from our overseas cousins. Creative Filing System, mentioned in the last issue, is now into version. 6.0.

Funnelweb, previously known as Funlwriter, is not changing its version number, tut there is a long list of improvements. including the ability to PRINT the disk directory from the Editor's SD. And you no longer need to re-enter the filename you want to load — just mark it in the SD directory and the none will appear when yo use LF or the Formatter. Then there is the beep to warn you of the approaching right hand edge. I could go on!

CREATIVE FILING SYSTEM is on THREE DISKS. FUNLWRITER is on TWO disks.

I now have some disks from FRANCE, including a good Forth game called BILLARD ... that is, NEARLY Billiards. It just doesn't have any pockets! Good game.

And an interesting item ... the source code for the TI-Writer editor. Not quite the module version, but nicely commented. TWO DISKS.

Remember the MACRO ASSEMBLER called RAGMAC is now on THREE disks! while c99 v2.1 is on FOUR disks.

I also have a separate c99 "tutorial" disk, but it is really a collection of functions, on the lines of the more you see the more you learn...

The DISK LIBRARY CATALOG is now over on to TWO disks - just send two disks and return post and packing and a copy is yours. And while sending disks, why not fill them with your latest works!

Keeping count of library programs is impossible! Even keeping track of the disks is becoming hard. Well over 200 disks at last count, and growing! And there are some GEMS in there!

The costs are low. Send in your own disk and it is just one pound per disk plus an extra one pound per package - which makes it cheaper per disk the more you order! The library operates on a non-profit basis, with any small surplus going to fairware authors.

Don't be shy. Write reviews about library contents that please you!! Share your experiences! WRITE!

An excellent range of offerings if you program in Extended Basic - these are for TI XB only, not Myarc:

JBM103 is a true bit map utility which lets you use true bit map graphics in your X6 programs, and, nice touch, you can save and load displays in a format which is compatible with TI-Artist.

STAR by Michael Riccio of COM-LINK Enterprises. A full range of CALL LINK utilities for TI Extended Basic. If you program in TI XB or would like to improve existing TI XB programs, you need this disk. 53 CALL LINK routines, including: screen save/load, bye, new, quiton, quitoff, charset without color change, title screen large caps, true lower case set, chimes, flashing text, VDP peek and poke, screen display on and off, read and write array to screen, instant sprite start/stop, check to see if ALPHA LOCK is engaged or SHIFT/CTRL/ FCTN keys are depressed, character copy, magnify, rotate, flip, mirror, invert, disk file protect/unprotect, disk catalog (not ramdisk), file rename, read and write sectors, 40-column mode plus 40-column PRINT, string reverse, and change strings to all capitals. Phew. PLUS SAMPLE PROGRAMS!

You want more? OK you have more. Still more routines for XB programmers:

ENHANCED DISPLAY PACKAGE from Paragon Computing is a true freeware package, keeping to the original ISM software of "you get what you pay for". From the library you get a disk with the object code, condensed (only 245 sectors!) documentation, and a DEMO program which shows you how to use most of the commands. Send \$10 and be kept up to date on enhancements and any

programs which use the package.

Send \$20 and get that plus full documentation. Send \$50 and you get that plus the source code. The utilities you XB programmers add to your computer include an ALARM CLOCK, windows, 40-column Mode with full handling commands, which are enhanced for both 32 and 40 columns: ACCEPT, DISPLAY, GCHAR, HCHAR, COLOR, SCREEN, VCHAR. Added features are GTEXT (!!), SCROLL, FLASH, enhanced CHARPAT and CHARSET, READ and WRITE screens, and so on. Quite a package. In 1983 this would probably have sold for \$50 plus, but it's yours for a lot, lot less.

----- THE PRINTER'S APPRENTICE

Update: In the last issue of TI*MES I reviewed THE PRINTER'S APPRENTICE. After more time with it, I have been able to produce a very acceptable page of mixed graphics and text in various fonts, and think it an excellent buy for the serious "page-maker" with a little time to spare!

I have also purchased an easier alternative, FONT WRITER by J. Peter Hoddie, which is sold by Asgard Software. The price is \$25 and credit cards are acceptable. Asgard Software: PO Box 10306, Rockville, MD 20850, USA.

Font Writer allows you to use TI-Artist fonts to print TI-Writer text, using the formatter dot commands (.CE and .FI for instance), adding some new dot commands to enable you to mix graphics into the text from TI-Artist Instances. CSGD images may also be used. It does not seem to be possible to mix graphics to the same extent or with the same control as with The Printer's Apprentice, which is cheaper at USD \$23! However, Font Writer requires less preparation and is written in Extended Basic, using Assembly Language routines. Font Writer is also even slower than The Printer's Apprentice.

For really good text/graphics my choice is TPA, but to quickly run off some text (quickly?) in fancy fonts, with perhaps the odd image here or there, Font Writer has its points.

In answer to some queries, I have no plans on ceasing the disk library operation! But, as with everything else, its existence does depend on support! And that support is coming from quite a small number of people right now! It will not take much of a decline to make acquisition of additional material a thing of the past.

----- OVERSEAS GROUPS! ATTENTION!

We seem to be a little light on newsletters from abroad. Attention! Do you know of any CURRENT group, publishing a WORTHWHILE newsletter who we could approach for an exchange of information? A sample newsletter would be helpful.

Unfortunately, due to the high cost of producing TI*MES it is not practical for us to send a copy of our mag out in exchange for a single page giving the minutes of the last meeting of a small group! But there are some equally large and informative newsletters cut there...

Basic Compiler

I have been waiting a while for somebody somewhere to REVIEW or even MENTION the Basic Compiler sold by Ryte Data of Canada, and advertised in the last issue of TI*MES. Looks like I'm going to have to find out for you! Order posted by airmail to Ryte Data on Tuesday, May 26. Advertised price was US\$22. MASTERCARD (Access to you!) accepted, so I'll use that! Keep yer eyes peeled for more news.

=====

XMLLNK : SCROLL

It is a frequent complaint of machine code programmers that the Extended Basic environment is hostile, but checking out the Editor/Assembler manual, it is apparent that with Extended Basic, there are a few additional utilities, to make up for the GPLLNK and DSRLNK which TI failed to supply us with - but which are now available in the public domain anyway!

A good example is the XMLLNK command: SCROLL. In a machine code program running in the Extended Basic environment, all you need do is:

```
BLWP @XMLLNK
```

```
DATA SCROLL
```

and off you go.

To do the same thing in an the E/A environment you must write your can routine. To see how much work TI have done for us, let's take a look at the SCROLL routine. The code below assumes that Workspace is at >83E0:

```

SCROLL LI R12,>02E0      (...CONTINUED)
      LI R10,>0020      AD  MOVB @>8800,*R7*
      CLR R9            INC  R10
      MOV R11,R6        INC  R8
      BL @AA           DEC  R12
      LI R5,>8C00        JEQ  AC
      LI R4,>02E0        ORI  R9,>4000
      LI R1,>7F80        CI   R9,>000C
      LI R2,>001C        MOVB R9,*R15
      BL @AF           AE  MOVB *R7+,@>8C00
      MOVB R1,*R5       INC  R9
      SWPB R1          DEC  R8
      AB  MOVB R1,*R5   JNE  AE
      DEC R2          MOVB R12,R12
      JNE AB         JNE  AA
      SWPB R1        B   *R11
      MOVB R1,*R5    AF  MOVB @>83E9,*R15
      MOVB R1,*R5    ORI  R4,>4000
      B *R6         MOVB R4,*R15
      AA  CLR R9      NOP
      MOVB @>83F5,*R15 MOVB R1,@>8C00
      STWP R7       B   *R11
      MOVB R10,*R15

```

That code came from DISKASSEMBLER, and looks a little odd to my inexperienced eye, but it indicates that writing for an XB environment, TI have cut down some work for you.

Correction

TI*MES 16. Page 16. Machine code string search: The code as given is correct for Extended Basic, but if you wish to run it from Editor Assembler, a further change is required: ExBas: CIF EQU >20 is correct but for Ed/As use: CIF EQU >2300

Thanks to Ron Johnson for pointing this out!

=====

XOP?

Ron also suggests that there is perhaps a need for console routines to be better documented and perhaps more samples of use, with discussion on various ways of tackling the same thing discussed too.

We long ago went through Basic with a toothpick, to see which ways used less memory, or ran fastest, but no-one seems to have done this for source code. Here's a challenge to you machine code programmers! Get dug in and share your results with us.

And has anyone used XOP, and does it have any advantages? Please write in.

=====

Jim Peterson has kindly sent on a few more TIPS FROM THE TIGERCUB and quite a few articles. Yes, I know the last edition said it was the last, but Jim still had some writing unwritten! Remember, Jim has lots of programs on commercial sale as well.

The April issue of MICROpendium has the price wrong too. Jim is not selling full disks off for \$2 - the price is \$10. Jim offers three "Nuts and Bolts" disks for \$15 EACH. Altogether these have 348 small subprograms for you to MERGE into your XB programs and use. There really is something for everyone here! Jim also offers, at \$10 each, four disks of TIPS FROM THE TIGERCUB, which have all 41 issues of Tips published up to January 1987. Text is in D/V 80, but all those programs are in runnable format.

Why not drop Jim a line - his address is in every Tips - and send some dollars.

=====

At Derby I received a copy of Myarc XB v2.12 (the third revision, and all free!) and APPEND now works!

For the money, this really is an excellent buy, and the 128K card you need to run it is probably more creatively useful than a second disk drive!

I also re-checked the benchmarks from TI*MES Issue 15, and v2.12 is the same as v2.10 as far as timing is concerned

=====

I don't write a lot on Basic things 'cos I have already written most of what I know in my book. There are lots of books still around on Basic, and NOBODY writes to me with questions on Basic. Hamm. Come to that, no-one has written to me about much in the last 4 months. IS THERE ANYONE THERE? Writing to a vacuum is remarkably hard! We don't have commercial market research firms working for us here. To know what you want, YOU have to tell us!

My address again:

10 Alstone Road STOCKPORT
Cheshire ENGLAND SK4 5AH

---if I repeat it enough maybe my reader will write...

+++++

PRINTER CODES

Page 40- Printer Commands - correction. The column for the FX80 MAY be for the FX80+, but it is NOT for the FX80! My FX80 does NOT support NLQ, but it DOES support PICA. The characters for Pica are 27-80.

The code for CONDENSED is incorrect- the FX80 (and I am sure the MX80) uses just 15 (no preceding 27) with code 18 for condensed off. !!!!!!!!!!!!!!!!!!!!!

WHEN DID TI QUIT?

Did I hear you say November 1983? Hmm. Perhaps. If you look at the bottom of your modules, you will find a code number made up of three letters and four numbers. The three letters indicate the factory- ATA=Austin LTA=Lubbock and so on. The next two numbers are the week number of manufacture and the final two numbers are the year of manufacture.

What is the LATEST module you have? I am still buying modules here, and have just bought one with the very clearly marked code of LTA1685. Can anyone beat that?? And looking at module codes, lets also look at CONSOLE codes, which follow the same format. We have found that consoles from 0182 to 0682 fail to function with several modules - notably the early Atarisoft, but some TI modules are also involved, yielding odd Japanese like characters instead of text or numbers. Can anyone extend the range of codes involved with this bug?

Also, according to The Smart Programmer, consoles made from 1281 to 3181 have sprites which flicker when moving vertically off screen - when one sprite moved vertically off screen, all lower numbered sprites flickered!

Any other oddities we can trace!

=====

Back to Issue 15 now...

THE ORPHAN SURVIVAL HANDBOOK

was advertised by Disk Only Software, a daft name for a book publisher, but still. I have a copy, and a good read it is too. A good collection of information on all sorts of things, including a very handy single page summary of the CODED error messages your TI can throw at you in various ways...

Some nice articles on the P-Code system and on C-99. Chris Bobbitts article on the Geneve, which seems to have been published EVERYWHERE! Some Basic programming tips and examples, a section on machine code, Forth, Hardware, TI Writer, Telecoms, and generally useful stuff. It comes loose leaf so you need to buy a binder! And as it is drilled for the US 3 ring binders, you need to redrill for our 2 or 4 ring binders.

Ooops... I see I said some nice things in issue 15 about MG and Byteaster. Since then MG has all but dropped the TI, and Byteaster is DECIDEDLY late with the SMART PROGRAMMER. Current advice: don't send any money to Byteaster for a while, the signs are you may get nothing for your trouble...

IMPORTING

Report from John Bingham that a firm called PILGRIMS PRIDE have cashed his cheque for US\$109 but NOT sent the goods and FAILED to respond in any way to his letters of enquiry. BE WARNED.

Traders you CAN trust: TENEX and TEXAMENTS.

(Triton refuse export orders). ++++++

!!!!!!!!!!!!!!!!!!!!

Tony McGovern has sent in quite a bit of text for us to publish, and has some harsh things to say about some folk. I don't like to take sides having only seen one viewpoint but as I reported in issue 15, things are getting positively DESTRUCTIVE out there in TI Land. Come on you chaps... cut it out, there aren't THAT many of us left these days! A cry similar to the above paragraph has also been published by the Ottawa group. The Chicago group have their own resident shredder who takes a page of each newsletter ripping into anything and everyone... and the LA group have made some fairly strong comments...

If there is anyone taking advantage of the orphan community, exposure is in order - but lets be sure of who is doing what, why, when and where. And lets be slow to criticise and swift to praise huh?

And I've written along similar veins to Richard Mitchell of Byteaster, and heard NOTHING in reply (not even a copy of the Smart Programmer...).

Aaaaaaaaghhh!!!!!!

Hmm.

????????????????????

Just had a letter from Christina asking for text NOW so not a lot of time to develop anything much! which will leave room for the articles from Tony and Jim - maybe!

+++++

WHAT'S HAPPENING...

I have just received three parcels from the USA, sent by AIRMAIL on March 23rd, April 17th and April 27th. The latter two had labels affixed apologising for the delay which was due to industrial action by HM Customs - and which is still continuing.

It seems unlikely I will receive the Rye Data Compiler before this issue goes to press, as from a Daily Telegraph report CANADA is one of the mail sources worst affected. It is reported that 250,000 packets are held up, some since February....

NEW LANGUAGES...

Two new languages, and both available from reliable sources...

99 FORTRAN US\$50 from TENEX

TURBO PASC 99 US\$80 from TEXAMENTS

Both prices exclude post. Both suppliers are reputable and rapid. Only Tenex take credit cards.

Do we have any members using similar languages already who would be prepared to order a copy of these new languages and report on them? Please?

Turbo Pasc does NOT require the P Code card and is said to be faster. TENEX: P O Box 6578, South Bend, IN, USA, 46660

-----> CONTINUED ----->

If using Barclaycard, say you are using VISA!
If using ACCESS, say you are using MasterCard!!
Indicate seairmail or airmail! Expect to pay about US\$10
for postage.

TEXTAMENTS: 53 Center Street, Patchogue, New York, USA,
11772

No credit cards- send International Money Order, Bank
Draft or (risky!) cash in US\$ only. Add \$15 for post.
Apart from Customs delays, these two suppliers usually
deliver in two to three weeks. Expect to pay the
postman about 22% duty and VAT - except that if the
strike action continues you may find your package
slipping through duty free!

::::::::::::::::::::::::::::

News from Chicago that John Behnke has been seen using
the RYTE Data Basic Compiler, advertised in our last
issue - watch out for a review by ME innext!!!

+++++ I understand the new Not Polyoptics
machine code flight simulator SPAD XIII, listed at
US\$30, is being sold by ASGARD for US\$28 plus post.

This is a pretty good simulation, in "real time", which
means that if it takes 9 minutes to fly to the Eiffel
Tower, then it takes 9 minutes to fly to the Eiffel
Tower!! The SPAD is an ancient plane which is very
stable in flight and hard to crash if you are high
enough to recover from any mishaps YOU might make!

The program includes simple flight simulation, flying
around the afore mentioned Eiffel Tower, zapping enemy
positions, dog fights, bombing enemy air fields.

Graphics are NOT bit-map. Unlike IBM we don't have
128k to play with, just 48k! so VECTOR graphics are
used - everything is plotted as dots which the computer
joins up with lines. Thus the sun is a large asterisk
and clouds are parallel white lines.

Views in 6 directions are available. Plane control is
GOOD - I managed a safe landing on my second attempt.
Not a bad program at all.

The disk is WELL copy protected.

ASGARD have so far sent me everything by seairmail,
regardless of how much I send for postage, so an extra
\$5 for post should be more than enough.

All credit cards.

ASGARD SOFTWARE, P O Box 10306, Rockville, MD, USA,
20850.

Seairmail can take two months or so! See if you can find
a US contact to buy for you and send airmail!

+++++

TI WRITER TIP

from Fox Cities US through Chicago...

Your cursor is flashing on the top line (the COMMAND
line) of TI Writer and you wish to return to EDIT mode.

How do you do it?

Like me? - Put an E up there and press ENTER?

There's an easier way... just press CTRL and Q and you
will jump straight to Edit mode.

I have searched through the TI Writer Manual and I
can't find any mention of this one...! But it WORKS.

=====

Lots of new fairware programs are being announced right
now, and as soon as funds allow I'm getting them in, so
keep in touch, and every time you send disks in for
recording, send in two more for a free up to date copy
of the CURRENT library list! OR just send the two disks
and return postage!

10 Alstone Road STOCKPORT Cheshire SK4 5AH

AND

Please write in! Questions for published answers,
'suggestions (detailed!!!) for topics to explore
together, NEWS and hints and tips to pass on.... are
you there!

////////////////////

'til next time happy TI-ing!

Stephen Shaw

ENTOMOLOGY CORNER No. 11

By Tony McGovern.

Edited by S Shaw.

There isn't really all that much to report so most of
this month's ravings are in the accompanying article on
the workings of TI-Writer's Editor.

The most recent update was with Vn 3.5 of DM-1000 when
the source code finally arrived last week. This has
been worked over to the same effect as with Vn 3.3
earlier and will find Horizon RAMdisks at CRU bases
>1200 and higher. I have sent the details back to
Ottawa and I finally hope that they incorporate it for
once and for all in DM-1000. The Vn 3.5 DM-1000 itself
has some new features which will be welcome, and some
bugs have been fixed.

An evil, insidious bug lurking in the central menu
screen loader has also finally been exposed and
squashed. This was most noticeably causing problems
with the loading of the c-Compiler from the menu screen
- sometimes it would and sometimes it wouldn't. What
the bug was doing was to interfere with the flag that
called for the loading of the E/A utilities, and the
compiler would load only if these were already present.
While I was at it I revised CD slightly so that it now
meshes in completely with REL3 (compiler Vn 2.1) of
c99. Also if CD,CP,.. are loaded by name from the
loader screen, it now knows not to try returning to FNB
which is no longer there, but still saves the filename
in the mailbox.

Another bug that turned up very recently crept in
during the change to Vn 3.4. Nobody has reported it
yet, and it only makes its presence felt when SD is done
on a disk with 100 or so files on it. It doesn't hurt
the Editor but crashes the machine on return to FNB.
Just a pair of pointers involved in juggling memory
that had escaped updating. Issues of FNB carrying
dates of Mar 87 or later will have this corrected.

We have also loosened the criteria by which FWB decides whether there is a valid name in the mailbox to be passed on to utility programs. Now it inspects only the first 3 characters for "DSK" or "RD." which allows disk volume name specification, or the universal name for the Myarc RAMdisk, the one it always responds to no matter what drive it is emulating.

Also filenames for the utility loaders may be up to 23 characters long. With a bit more foresight this should have been 25 but that will have to do until the next major revision if that ever occurs. This means that volume name access may be used for utility files. The longer names may also be used in the LOAD program for the XB level of user list.

If volume name specification of assembly language utilities is used here then the auto boot-disk tracking must be disabled by the K value for that entry. This is because boot disk tracking and volume name specification are incompatible in principle and practice.

The User List accessed from the central menu screen and defined with ULINSTL still only accepts 15 character names, and this cannot be increased because there just aren't enough bytes available.

The 'fairware' system struggles along, maybe as well as one could expect in the TI market. Maybe us smalltown folk are right in thinking that big cities are bad places. Not a single response from the whole of Sydney, and just one from the whole of the Greater Los Angeles area, other than someone making a fairware survey. Nothing is black and white here, but the TI-99 market is one where even commercial software on the whole has been very reasonably priced.

Not like for other machines where absolutely exorbitant prices of software subsidise massive advertising and brainwashing campaigns, battalions of lawyers, and layers of high margin resellers.

Will has been extremely disappointed with the response to Disk-hacker, from a few established friends and the odd previous stranger or two. This is despite evidence of frequent downloads from USA networks. That is the reality of the 'fairware' system. Readers of this series have probably noticed that I have a distaste, frequently expressed for hype.

There is still too much of it around, but there are fortunately solid and well designed programs out there that will help prolong the life of our computer investment. I would be quite disappointed to find that anyone competent to use Funnelweb didn't find it as least as good as I have ever made it out to be.

The name 'Funnelweb' and derivatives have always been our private joke which occasionally causes some comment. Just as well names of programs don't bite.

At the moment we are saving up for a MYARC 9640, influenced perhaps as much as anything by the investment in time spent on and liking for 9900 family assembly language. I am also aware that the computer board is only a small part of what is involved, because if you are not going to throw away most of the graphics capabilities of the 9640 you are going to be in for the cost of a color monitor with far better resolution than the television receivers which are adequate for the 99/4a.

William still wants an Amiga, but is realistic enough to wait for new models of that to appear. My own assessment is that if we were to get an Amiga first, we would get so involved with 68K coding that we wouldn't bother with the 9640 and the 99/4a would be relegated to being a backup machine for routine work with no further program development being of interest. The 9640 and 99/4a would have a lot more in common for development purposes if we did have the new machine.

In fact the updated 99/4a equipped with 2x360K drives and RAMdisk(s) is still an excellent computer, 40 column display and limited keyboard notwithstanding. If Funnelweb is used as the operating system shell then it loses very little in comparison with other machines over ease of use either.

You should also ask your friends with other machines what happens to the contents of their RAMdisks when they have program crashes and lockups. Whatever happens your TI-99, with programs like FWB and Clint Pulley's c99 you can get in practice for moving on to newer machines.

There is a subtle trap with the TI-99 in that once you get involved with TMS-9900 Assembler, you find that it is the most elegant for any micro that I know of and a real pleasure to write as such things go. The trap is that the TI-99 is a small enough machine that you are not forced into higher level languages by the tyranny of program size, so I never seem to get around to p-code or c99.

(Tony has not sent me No 12 - sjs)

ENTOMOLOGY CORNER No. 13

Tony McGovern

Funnelweb Farm
(Edited by S Shaw)

We bought our expansion system just after TI sent us all into the orphanage, because it looked the best buy around in a home computer system, a view amply confirmed in the years to come. At the same time we ordered E/A and TI-Writer, and they duly arrived (E/A with a dud disk which took some time to replace) at a cost of about \$50 apiece, which was a bit lower than they had listed for previously.

----> continued ---->

TI eventually sold out its stock in a bulk deal to Iagac Australia, who apparently figuring they had the market in the proverbial grip, raised the prices of E/A and TI-Writer to \$150 and \$200 and more. This was the background to the development to the first TI-Writer only version of what has grown into Funnelweb.

This saved some of our members from the extortioners over TI-Writer, but in the time between then and the appearance of Vn 3.0 which supported E/A functions, some, anxious to get into Assembler, did end up paying out the fortune demanded for E/A.

[Tony wrote at some length here about Network Nasties, some pretty foul comments on himself which appeared on U S bulletin boards - where he was not supposed to see them! Principal authors were Howie Rosenburg of Long Island and Richard Mitchell, publisher of The Smart Programmer.

As you haven't seen the text, and Tony did not waste space reproducing it, I have cut his comments. Suffice to say that a couple of US owners have a bee in their bonnets about Funlweb and Tony...]

Enough of the Network Nasties. The only good thing to say about them is that that they gave something to write about for this month's article. After the effort on TI-Writer last month I was wondering whether there was anything left to say. As a result of the Nasties I have included a paragraph in the -READ-ME file of the FUNNELWEB package forbidding the placing of the Funnelweb files on any bulletin board or information network without our explicit permission. This will normally be immediate for non-profit BBS's associated with User Groups, but for commercial networks the request will have to come from a commercially and legally responsible official of the company.

Continuing our usual look around the TI-99 scene, the Micropendium seems to arrive in the hands of Newcastle subscribers as regularly as the mail from Texas permits. The scope and quality of articles as always are variable, but there is usually something of interest for all levels. At least the subject of direct disk controller access was introduced there.

At this stage I have to apologize for going off again over the attempts to silence critical reviewing coming from some quarters. I'd much rather be talking about technical matters and doing the reviews of the TI-99 scene, or just plain gossiping, but an answer seems required to the latest. This was in the form of a so-called Open Letter by Tom Freeman, which has come to us indirectly. Again such of it is in reckless disregard of the truth.

The first part is a less than complimentary discussion of "fairware". Now my impression is that the writers of the top fairware items are quietly pleased with the response to their efforts, even if they aren't think about it in terms of cents per hour return.

I think the background is that the MG/SP case is perturbed by the situation that has developed where very many users have found their needs adequately catered for by their better TI-original software, general purpose hardware updates, and fairware.

As far as I can see the only area where application software is dominated by commercial programs is in graphics utilities, an area where MG is not prominent, despite their name. Once TI left the field, I suspect that differences between commercial and fairware program writing became very small, the work being done on TI-99 systems at home in evenings and weekends over months and years of obsession.

The main difference may have been the amount of TI internal documentation available to various programmers. I presume the leakage paths were shortest in the US. On the whole, the self-motivated and self-driven fairware programmers seem to have met the needs of the user community best.

Perhaps it is all the effort put into designing protection schemes that bring no benefit to the user that makes the difference. Freeman also complains about frequent updates to fairware programs.

Now I know that revision of Funnelweb is always a standing joke around HV99, but we made a decision early in the piece that the spirit of fairware was best fulfilled if bug-fixes and improvements were to be shared as soon as they were available, and would feel very uncomfortable if this weren't so.

I think pretensions to perfection or finality in complex programs are just that, and it is the continuing development of fairware, and the stagnation of some commercial programs that proves it. If a complex program does what it does and what is claimed or implied for it perfectly, and very few do, then that is usually a sign that it could do more things better. I sometimes get letters from people who are immensely happy with Vn 3.1 of Funnelweb and are surprised to learn that it has moved on to Vn 3.4 with more work in the updating than in the original.

Now we come to the parts where Tom seems to run off into his own private horror fantasies, lashing around to fit up Will and me with these in reckless disregard of the actual facts. It should be a good laugh for HV99 members to hear Will described as "precious".

First I think we should set a few things straight, seeing as Freeman has, like the Network Nasties, been so persistent in misrepresenting the written record. This does seem incredible, but the written record is there for anyone who wants to take the time to read it. Firstly we do own one of the MG programs, DISKASSEMBLER which was a Xmas present, and we have used the other two on brief occasions from borrowed genuine original disks. The occasions were brief because despite Freeman's comforting words, all the owners of these protected disks were at the time extremely nervous about their investment and its fragility.

RAMBLES

From what we saw of those two they were not something we would use except on rare enough occasions that we could borrow them. That's the way it has worked out too, and they were useful only as track analyser test material. My estimates on DKA which led me to try to buy a copy have proved too optimistic.

Since we have had it here I've only used it to reconstruct the patches I had made to EDITA1 after losing the file that recorded them, and I can't really envisage much future use.

These patches had originally been done with the QIXOFT disk manager, an excellent program from that most solid citizen of the Australian TI-99 scene, Bernie Elsner, and Phil West. [Stephen here - a citizen with two names haaa! No, BOTH good citizens!]

Now that had a user interface for the sector editor that we liked much better than what we saw of AD. These days we always use the reworked DPatch that is an integrated part of Funnelweb.

Though the MG protection schemes proved an interesting but shortlived technical challenge for Will we have always been very careful to respect MGs business and to let their products sell on their merits. Freeman's apparent attitude that users may not possess a back-up of their uncopyable disk except on MGs terms is user-hostile.

I would say that those very few of our friends who have a backup now for their original, have relaxed a little because they are no longer dependent on one fragile disk from a supplier with that attitude.

Another aspect came to light in a letter received today from a TI user who has recently installed a hard disk system and finds that he cannot use his set of MG programs on the hard disk.

That's a tougher problem than just backing up a disk. It has happened with protected software on other machines and generally it has just been tough cheddar for the hapless customer. Pirates are only the minor league of brigands in the software business.

All I can advise him in principle is to contact MG and hope he has better luck than I did. If not then I would advise him as a practical matter to seek out quietly the pirate versions, made and circulated in USA, and as he is already a paid up owner, not to feel guilty about it given the present evidence of the kind of people he bought them from originally.

Come to think of it, the MG camp followers will on exhibited form, extract phrases out of that last sentence for selective misquotation just as they did out of the letter that Will wrote to his supposed young friend in the USA.

I have in print recommended the MG programs (and most commercial software for the TI for that matter) as

being low enough priced compared to that for other machines that piracy could not be justified in any way. I practise and preach the attitude that one should refuse to purchase copy protected software.

This level of user revolt has become widespread enough on some other machines that copy protection, which in some cases has been far more obnoxious than on the TI has gone from being a hot subject to a dead issue.

Now in a TI-99 household where the programmers get their kicks out of writing operating system shells for a machine that was intended never to have one, or from making disk controllers jump through hoops, not using these programs is less of a hardship than might be imagined.

What is particularly obnoxious about the MG camp's current effort to silence criticism of their products by abuse and slander of those who offer any, is the attempt, deliberate from appearances, to twist that expressed attitude of ours around to make it look like one of condoning and even being active in piracy of MG products.

Next Tom rallies to the defense of MGs Adv Diags. Now we criticized it on the very specific grounds that its much touted track reading feature in fact just doesn't do that properly. Somehow Freeman translates that specific criticism into a general prejudice, a reaction perhaps to be expected from someone who chooses not to use our program because he says he doesn't like our "policies".

Well I am prejudiced against products which don't deliver what they promise, and I think our User Group members deserve to get an incisive review, rather than mere rehash of promotional material.

Freeman's admission of the obvious, that AD deliberately conceals information about the low level capabilities of the TI disk system, no matter how it was done in detail, confirms our comments in review and does not mesh well with the carefully cultivated image as a producer of products to help people understand their computer.

When it comes to DISKHACKER, Tom seems mainly to be reacting to the teenage bravado of the name. Unlike the various track-copy programs of US origin that allow backing up of most protected programs without the user having to understand what was on the disk, DHR analyses the disk contents for valid sector/track structures, and presents the analysed results.

It does NOT in fact provide any copy function. Now that is some sort of educational program.

I expect also, schoolwork permitting, that updates will appear if the vicious personal attacks don't sour him on the TI-99 altogether.

Next comes the part more directly a reaction to our comments on Freeman's own programs. The intemperate invective here is more understandable but no less inexcusable for that.

Some of it is not far removed from the "Have you stopped beating your wife?" category. The club librarian handed me the latest SP to arrive the other day. Curiously one item in it is that a formerly protected MG program, Explorer, the most impressive programming effort of the bunch by a country mile, is to be offered in unprotected form.

Perhaps all this smear campaign by the MG camp followers is merely a smokescreen to cover a tactical retreat. I know that Americans by and large take a very serious view of life and can find the Australian sense of humour quite bewildering, but it almost defies belief that someone could take a facetious remark about paper consumption in printouts as an attack on their personal integrity. I guess it reflects the way our open letter writer thinks.

Facetiousness aside it was comment on a program design that did not consider economy in printout. I guess I didn't realize it was all so perfect it was beyond critical comment.

The most curious aspect of it all is that a private letter written by Will to a youthful programmer in the US seems to have become the public property of a whole bunch of very unpleasant people.

Will is steaming mad about it, not so much that his personal enthusiasms of the time have become a public peep show, but at the betrayal of trust by someone he thought was a friend.

I don't see that he can be 100% sure of his interpretation of events so I've said to the senior member of that young person's User Group with whom we correspond regularly that I would not identify them in any way. I'm annoyed because I want Will to write in a language other than TMS-9900 assembler and that's less likely now.

Now peeping Tom seems to take teenage enthusiasms of the the moment as true records graven for all time. While Will was tackling protection schemes I used to be hit with this barrage of ideas spilling out on how a particular protection might have been or was done, and of the ways that he could then beat his own latest cracking methods.

I would patiently remind him that protection was a counterproductive waste of time anyway and that anything he could think of would just as easily be broken by his US high school counterparts. Eventually he would cool down from the white heat of his detailed technical enthusiasms of the moment.

Will also makes a habit of loading every new and different program with Funnelweb and takes fiendish delight in finding ones that don't or in any other bugs he can turn up. Suffice to say that Funnelweb and its predecessors have never been issued with MGs precious programs on board.

The document files are quite explicit in that area. The only MG program to reside on our private working Funnelweb disk is the E/A version of DKA. Our attitudes on that are purely pragmatic - it is the least used program on the disk and will sooner or later be displaced by something else we want to use more often.

Please don't write to me requesting E/A program file backups of your MG protected disks. Commercial distribution of MGs programs is and always has been MGs business.

Just write to your dealer or to MG directly, and if you are not satisfied with their response make your feelings known in your User Group Newsletter or whatever other forum you have available. They or their associates can't possibly attack everyone with a campaign like the present.

I am rather surprised at the description of me as a hacker. Perhaps I should be flattered that someone thinks my aging brain is that agile. Perhaps there is hope for my second childhood.

Will isn't going to get too far along that line either as we don't have a modem and couldn't afford the phone bills anyhow. Maybe the original uncorrupted sense of the word hacker is meant, the skilful programmer working long hours to produce programming feats. Given the rest of the article is almost totally devoid of accuracy, that is unlikely.

All this furor has been a reminder that I may have been remiss in not reviewing fairware items more frequently and thoroughly. Any that I see are readily available from the club library for a nominal charge, without the money demanded up-front for commercial software, and review seems almost superfluous.

Commercial products are not accessible so easily, so if they pass our way I think it is a service to our members to comment on them. It may very well be that my perceptions are not those of the image that a particular producer is at great pains to project.

If that constitutes "Miller bashing" then so be it. Until now I have had no reason to think anything bad of the Miller operation and regarded them as a substantial net contributor to the TI-99 scene. Now I find personal difficulty in separating their image from that of the last thrashings of IUG and HCM.

I find it strange that no direct communication of any complaints about the comments on their products was ever received from MG. I know I'm not infallible in either programming or commentary, and will always attend to corrections to either as appropriate.

It would have been interesting to have heard directly rather than by second and third hand reading of untruthful, libellous diatribes. It would still be of interest to hear disavowals from people whose names might be associated with MG/SP who feel demeaned by possible association with the vicious smear campaign indulged in by some. □

LATE ITEMS:

The last issue of TI*MES mentioned the TRITON TURBO XT using a TI99/4A Keyboard. This product was thought so silly it even got a mention in the Chip Chat section of the July 1987 issue of PERSONAL COMPUTER WORLD - complete with the first photo of a TI99/4A to appear in PCW for quite some time...

CorComp have sent in their latest product news, and Yes, another IBM type product, described as:

The "Real" TI/IBM Connection - but headed more rationally as:
Revolutionary TI/IBM Disk Copier.

What it does: Reads your TI Writer text disks and then writes the text out to a disk formatted in a format an IBM PC can read... thats it.

What you NEED: A CorComp disk controller AND a double sided double density disk drive.

To need it you need both a TI99/4A and an IBM. And an inability to connect them directly using RS232 locally or through a modem. No indication of the price. If you need it I suppose you need it - but I remember a long time back John Rice telling us he printed his TI Writer disks out using a BBC micro...

Another CorComp product of little value to many but of great value to some...

is a word processor!!! Called WRITEREASE the advert reads like a cut-down TI-Writer, but extras include a 30000 (thirty thousand) word disctionary !

The other extra feature is the ability to have column widths up to 255 characters wide. No price. !g

Back to IBM and PCW (July 1987)

"...I really do wonder just how many people in this country are equipped to learn MS-DOS or GEM, and I suspect that the number is lower than many pundits imagine".

The April 1987 issue of the Bayou 99 Users Group newsletter (vice president: Richard Mitchell) lists 12 suppliers of TI products. Personally I would avoid 8 of them like the plague! My favorite, Texaments is not mentioned!

Confirmed: MG (lately Millers Graphics and former publishers of The Smart Programmer magazine) HAVE stopped making GramCracker and NO-ONE ELSE is making it. MG have also stopped selling EXPLORER. However, Bytemaster (R Mitchell t/as, present publishers (query?) of The Smart Programmer) are now(?) selling EXPLORER in an UNPROTECTED version.

MOST POPULAR FREeware on Compuserve, according to COMPUTE! (via CinDay) is a terminal emulator (you need one to download it...). Then comes DM1000 and Funlwriter - these are both popular here! but Funlwriter first, then lagging a way beyond, DM1000. Fourth in the US popularity stakes comes NEATLIST, a program which lists Basic/XB programs very neatly- and in the UK, Neatlist has so far managed an ALL TIME distribution by me of....THREE COPIES!!!!

I have some really nice programs that NOBODY is looking at! Why not send for some and write about them for us! Be brave!

How to alter VDP registers (mini memory / ed/as basic):

A long time back I showed you how to use clever CALL PEEKV's to change VDP registers to do all sorts of clever things! Hmm. Guess what? You really DON'T have to use the illogical PEEKV at all, you can use the glaringly obvious POKEV instead. Its so long ago I can't remember WHY I suggested PEEKV!!!

TIPS FROM THE TIGERCUB

#42

Copyright 1987

TIGERCUB SOFTWARE
156 Collingwood Ave.
Columbus, OH 43213

Distributed by Tigercub Software to TI-99/4A Users Groups for promotional purposes and in exchange for their newsletters. May be reprinted by non-profit users groups, with credit to Tigercub Software.

Over 130 original programs in Basic and Extended Basic, available on cassette or disk, now reduced to just \$2.00 each, plus \$1.50 per order for cassette or disk and PP&M. Cassette programs will not be available after my present stock of blanks is exhausted.

Descriptive catalogs, while they last, \$1.00 which is deductible from your first order.

Tigercub Full Disk Collections, reduced to \$10 postpaid. Each of these contains either 5 or 6 of my regular \$2 catalog programs, and the remaining disk space has been filled with some of the best public domain programs of the same category. I am NOT selling public domain programs - they are a free bonus!

TIGERCUB'S BEST, PROGRAMMING TUTOR, PROGRAMMER'S UTILITIES, BRAIN GAMES, BRAIN TEASERS, BRAIN BUSTERS!, MANEUVERING GAMES, ACTION REFLEX AND CONCENTRATION, TWO-PLAYER GAMES, KID'S GAMES, MORE GAMES, WORD GAMES, ELEMENTARY MATH, MIDDLE/HIGH SCHOOL MATH, VOCABULARY AND READING, MUSICAL EDUCATION, KALEIDOSCOPIES AND DISPLAYS

NUTS & BOLTS (No. 1), a full disk of 100 Extended Basic utility subprograms in merge format, ready to merge into your own programs. Plus the Tigercub Menuloader, a tutorial on using subprograms, and 5 pages of documentation with an example of the use of each subprogram. Reduced to \$15.00 postpaid.

NUTS & BOLTS NO. 2, another full disk of 108 utility subprograms in merge format, all new and fully compatible with the last, and with 10 pages of documentation and examples. Also \$15 postpaid.

* NUTS & BOLTS #3 is now *
* ready, another full disk *
* of 140 new merge-format *
* utility subprograms, all *
* compatible with the pre- *
* vious. With 11 pages of *
* documentation, \$15 ppd. *

TIPS FROM THE TIGERCUB, a full disk containing the complete contents of this newsletter Nos. 1 through 14, 50 original programs and files, reduced to \$10 ppd.

TIPS FROM THE TIGERCUB VOL. 2, another diskfull, complete contents of Nos. 15 through 24, over 60 files and programs, also just \$10

TIPS FROM THE TIGERCUB VOL. 3, another 62 programs, tips and routines from Nos. 25 through 32, \$10 postpaid.

TIPS FROM THE TIGERCUB VOL. 4, another 48 programs and files from issues 33 through 41, also \$10 postpaid.

110 OPEN #1:"DSK1.BXBDATA",V
ARIABLE 163,OUTPUT :: PRINT

```
#1:CHR$(117)&CHR$(50)&"[ \ ]
$"&CHR$(190)&CHR$(199)&CHR$(
136)&M$&CHR$(0)
```

The Hyphenated Fill and Adjust in Tips #41 will crash if the file contains a line with one character too many, which may be only an unnecessary control character. This fix will help -

```
300 IF LEN(M$)<=L THEN 310 :
: CALL SOUND(200,110,0,-4,0)
:: PRINT M$;" is";LEN(M$);"c
haracters long":"Truncated t
o ";SEG$(M$,1,L):"OK? (Y/N)"
305 CALL KEY(3,K,S):: IF S=0
THEN 305 ELSE IF K<>89 THEN
STOP ELSE M$=SEG$(M$,1,L)
310 PRINT #2:M$ :: IF EOF(1)
<>1 THEN 220 ELSE CLOSE #1 :
: CLOSE #2
```

I know that this line is wrong, but key it in just as it's printed, and see what kind of error message you get -

```
100 !DISPLAY AT(3,1):"Progra
m must be SAVED in:"MERGE fo
rmat."
```

A friend asked me for a program to help him solve the Scram-Lets puzzles in our local newspaper, so I rewrote the Anagrammer that was published way back in Tips #12. It will print out all possible combinations of any 3- to 6-letter word, or only those which have one or two letters in specified positions.

```
100 CALL CLEAR :: DISPLAY AT
(3,5)ERASE ALL:"SCRAM-LETS S
OLVER": ! by Jim Peterson
110 DISPLAY AT(8,1):"OUTPUT
TO? 1:" (1) SCREEN": (2)
PRINTER" :: ACCEPT AT(8,12)
VALIDATE("12")SIZE(-1):P ::
P=P-1
120 IF P=1 THEN DISPLAY AT(1
2,1):"PRINTER? P10" :: ACCEP
T AT(12,10)SIZE(-18):P$ :: 0
PEN #1:P$
130 PL(1),PL(2)=0 :: L$(1),L
```

```
$(2)=" :: DISPLAY AT(5,1)ER
ASE ALL:"TYPE A 3-,4-,5- OR
6-LETTER WORD " :: ACCEPT A
T(6,6):A$ :: W=LEN(A$):: IF
(W<3)+(W>6)THEN 130
```

```
140 DISPLAY AT(14,1):"SEARCH
FOR COMBINATION WITH":"LETT
ER IN KNOWN POSITION? N" ::
ACCEPT AT(15,27)VALIDATE("YN
")SIZE(-1):Q$ :: IF Q$="N" T
HEN 180
```

```
150 DISPLAY AT(17,1):"LETTER
?" :: ACCEPT AT(17,9):L$(1):
: DISPLAY AT(19,1):"POSITION
?" :: ACCEPT AT(19,11):PL(1)
160 DISPLAY AT(21,1):"ANOTHE
R LETTER/POSITION? N" :: ACC
EPT AT(21,26)VALIDATE("YN")S
IZE(-1):X$ :: IF X$="N" THEN
180
```

```
170 DISPLAY AT(21,1):"LETTER
?" :: ACCEPT AT(21,9):L$(2):
: DISPLAY AT(23,1):"POSITION
?" :: ACCEPT AT(23,11):PL(2)
```

```
180 PRINT #P :: FOR J=1 TO W
:: B$(J)=SEG$(A$,J,1):: NEX
T J :: FOR J=2 TO W :: IF B$(
J)>B$(J-1)THEN 220
```

```
190 T$=B$(J):: FOR L=J-1 TO
1 STEP -1 :: B$(L+1)=B$(L)
200 IF B$(L-1)=T$ THEN 210
:: B$(L)=T$ :: GOTO 220
```

```
210 NEXT L
220 NEXT J
230 FOR A=1 TO W :: FOR B=1
TO W :: IF B=A THEN 440
```

```
240 FOR C=1 TO W :: IF (C=A)
+(C=B)THEN 430
```

```
250 IF W=3 THEN 310
260 FOR D=1 TO W :: IF (D=A)
+(D=B)+(D=C)THEN 420
```

```
270 IF W=4 THEN 320
280 FOR E=1 TO W :: IF (E=A)
+(E=B)+(E=C)+(E=D)THEN 410
```

```
290 IF W=5 THEN 330
300 FOR F=1 TO W :: IF (F=A)
+(F=B)+(F=C)+(F=D)+(F=E)THEN
400 ELSE 340
```

```
310 W$=B$(A)&B$(B)&B$(C):: I
F W$<=V$ THEN 430 ELSE 350
320 W$=B$(A)&B$(B)&B$(C)&B$(
D):: IF W$<=V$ THEN 420 ELSE
350
```

```
330 W$=B$(A)&B$(B)&B$(C)&B$(
D)&B$(E):: IF W$<=V$ THEN 41
0 ELSE 350
```

```
340 W$=B$(A)&B$(B)&B$(C)&B$(
D)&B$(E)&B$(F):: IF W$<=V$ T
HEN 410
```

```
350 IF Q$="N" THEN 380
```

```

360 IF SEG$(W$,PL(1),1)<>L$(1) THEN 390
370 IF X$="N" THEN 380 ELSE
IF SEG$(W$,PL(2),1)<>L$(2) THEN
EN 390
380 PRINT #P:W$&" ";;: G=6+1
390 V$=W$ :: ON W-2 GOTO 430
,420,410,400
400 NEXT F
410 NEXT E
420 NEXT D
430 NEXT C
440 NEXT B
450 NEXT A
460 PRINT #P: " ";;: "TOTAL
COMBINATIONS." : : : G=0 ::
V$="" :: PRINT "PRESS ANY KEY"
470 CALL KEY(O,K,S):: IF S=0
THEN 470 ELSE 130

```

And here is a much-improved XBasic version of the Adder-Upper which first appeared in Tips #13. I find it very useful in adding up several categories of figures in one pass.

```

100 CALL CLEAR :: CALL SCREEN(16):: FOR SET=1 TO 14 :: CALL COLOR(SET,5,1):: NEXT SET
110 DISPLAY AT(3,4)ERASE ALL : "TIGERCUB ADDER-UPPER": "To add up several categories": "at one time.": "Input categories - END when": "finished"
120 CALL KEY(3,K,S):: DIM C$(22),T(22)
130 X=X+1 :: DISPLAY AT(12,1): "Category #";STR$(X):: ACCEPT AT(12,13):C$(X):: IF C$(X)="END" THEN X=X-1 :: GOTO 170
140 A$=SEG$(C$(X),1,1):: IF POS(F$,A$,1)=0 THEN F$=F$&A$ :: IF X<17 THEN 130 ELSE 170
150 DISPLAY AT(15,1): "Code letter ";A$;" already": "used.": "Pick another code letter" :: ACCEPT AT(17,26)SIZE(1): A$
160 IF POS(F$,A$,1)<>0 THEN DISPLAY AT(15,1): ; ; ; ; ; : GOTO 150 ELSE F$=F$&A$ :: C$(X)=A$&C$(X):: DISPLAY AT(15,1): ; ; ; ; ; : IF X<17 THEN 1

```

```

30 ELSE 170
170 CALL CLEAR :: R=2+(X>8): FOR J=1 TO X :: DISPLAY AT(R,1): "(";SEG$(C$(J),1,1);" ";SEG$(C$(J),2,255):: R=R+2+(X>8):: NEXT J
180 DISPLAY AT(R+2,1): "Category ";F$ :: DISPLAY AT(R+4,1): "Amount"
190 DISPLAY AT(24,1): "Use minus value to subtract"
200 ACCEPT AT(R+2,11+LEN(F$))SIZE(1)VALIDATE(F$):Z$ :: Y=POS(F$,Z$,1)
210 ACCEPT AT(R+4,8)VALIDATE(NUMERIC):A :: T(Y)=T(Y)+A :: DISPLAY AT(Y*(2+(X>8)),20): T(Y):: GOTO 200

```

Can you figure this one out? (I can't!) -

```

100 DISPLAY AT(3,4)ERASE ALL : "ILLOGICAL COMPUTER!": " by Tigercub"
110 DISPLAY AT(7,1): "100 IF A=2 THEN IF B=2 THEN C=4 ELSE IF A=2 THEN IF B=3 THEN C=6 ELSE IF A=3 THEN IF B=3 THEN C=9 ELSE IF A=3 THEN IF B=4 THEN C=12 ELSE C=9"
120 DISPLAY AT(14,1): "Why can't you get C to ": "equal 9 or 12 or 99?"
130 DISPLAY AT(18,1): "A? " : : ACCEPT AT(18,4):A :: DISPLAY AT(20,1): "B? " : : ACCEPT AT(20,4):B
140 IF A=2 THEN IF B=2 THEN C=4 ELSE IF A=2 THEN IF B=3 THEN C=6 ELSE IF A=3 THEN IF B=3 THEN C=9 ELSE IF A=3 THEN IF B=4 THEN C=12 ELSE C=9
150 DISPLAY AT(22,1): "C=";C :: GOTO 130

```

This might come in handy to dress up a program -

```

100 CALL CLEAR :: CALL COLOR(2,5,16):: CALL HCHAR(1,1,42,768)
110 X=X+1 :: DISPLAY AT(X,9): "*****";: DISPLAY AT(X+1,9): "PRESS ANY KEY";: DISPLAY AT(X+2,10): "TO CONTINUE";
120 CALL KEY(O,K,S):: ON S+1 GOTO 110,130

```

```

130 !continue program here
Or, if you'd rather do it backwards -
100 CALL CLEAR :: CALL COLOR(2,5,16):: CALL HCHAR(1,1,42,768)
110 FOR X=10000 TO 1 STEP -1 :: DISPLAY AT(X+2,9): "*****";: DISPLAY AT(X+1,9): "*TO CONTINUE*";: DISPLAY AT(X,9): "PRESS ANY KEY";
120 CALL KEY(O,K,S):: ON S+1 GOTO 130,140
130 NEXT X
140 !continue program here

```

You might find this one useful -

```

100 ! PAINT CALCULATOR by Jim Peterson
110 CALL CLEAR :: FOR SET=1 TO 12 :: CALL COLOR(SET,2,8) :: NEXT SET :: CALL SCREEN(5) :: CALL KEY(3,K,S):: ON WARNING NEXT
120 DISPLAY AT(3,7)ERASE ALL : "PAINT CALCULATOR": "To determine the amount of": "paint needed for a room."
130 DISPLAY AT(8,1): "Is the room a regular square or rectangle? Y" :: ACCEPT AT(9,16)SIZE(-1)VALIDATE("YN")BEEP: Q$ :: IF Q$="Y" THEN 160
140 DISPLAY AT(11,1): "How many rectangular areas": "does the room contain?" :: CALL ACCEPTER(12,24,A):: IF A=1 THEN 160
150 FOR B=1 TO A :: DISPLAY AT(3,10)ERASE ALL: "AREA #";B :: GOTO 170
160 CALL CLEAR
170 DISPLAY AT(5,1): "How high is the ceiling?": " ft. in." :: CALL ACCEPTER(6,2,HF)
180 CALL ACCEPTER(6,9,HI):: HI=HI/12 :: H=HF+HI
190 DISPLAY AT(8,1): "How many walls?" :: CALL ACCEPTER(8,17,W):: CALL HCHAR(5,1,32,640)
200 FOR J=1 TO W :: DISPLAY AT(5,10): "WALL #";J: "Width ft in" :: CALL ACCEPTER(7,7,WF)
210 CALL ACCEPTER(7,13,WI)::

```

```

WI=WI/12 :: WW=WF+WI :: SQ=SQ+H*WW
220 DISPLAY AT(11,1): "How many doors, windows or": "other areas not to be": "painted in wall #";J;"?"
230 CALL ACCEPTER(13,19,D):: IF D=0 THEN 280
240 FOR L=1 TO D :: DISPLAY AT(15,1): "AREA NOT TO PAINT #";L: "Width ft in" :: CALL ACCEPTER(17,10,WDF)
250 CALL ACCEPTER(17,16,WDI):: WDI=WDI/12 :: WD=WDF+WDI
260 DISPLAY AT(19,1): "Height ft in" :: CALL ACCEPTER(19,11,HDF)
270 CALL ACCEPTER(19,17,HDI):: HDI=HDI/12 :: HD=HDF+HDI :: SQ=SQ-WD*HD :: NEXT L
280 NEXT J :: DISPLAY AT(21,1): "Paint the ceiling?" :: ACCEPT AT(21,20)SIZE(1)VALIDATE("YN"):QQ$ :: IF QQ$="N" THEN 320
290 CALL HCHAR(5,1,32,640):: DISPLAY AT(5,1): "Ceiling dimensions": " ft in by ft in" :: CALL ACCEPTER(7,2,CWF)
300 CALL ACCEPTER(7,8,CWI):: CWI=CWI/12 :: CW=CWF+CWI
310 CALL ACCEPTER(7,17,CLF):: CALL ACCEPTER(7,23,CLI):: CLI=CLI/12 :: CL=CLF+CLI :: SQ=SQ+CW*CL
320 CALL HCHAR(5,1,32,640):: IF Q$="Y" THEN 340
330 NEXT B
340 DISPLAY AT(3,1)ERASE ALL : "Total of";INT(SQ+.5);"square feet."
350 DISPLAY AT(5,1): "How many square feet will": "one gallon of your paint": "cover?"
360 ACCEPT AT(7,8)SIZE(3)VALIDATE(DIGIT)BEEP:SF :: DISPLAY AT(9,1): "How many coats?" :: CALL ACCEPTER(9,17,C):: G=SQ/SF+C :: G=INT(G+.5)
370 DISPLAY AT(15,1): "You will need";G;"gallons or": G*4;"quarts of paint."
380 CALL KEY(O,K,S):: IF S=0 THEN 390 ELSE STOP
390 SUB ACCEPTER(R,C,Q):: ACCEPT AT(R,C)SIZE(2)VALIDATE(DIGIT)BEEP:Q :: SUBEND
Memory full! - Jim P.

```

TI-WRITER UNDER THE HOOD

By Tony McGovern.

Have you ever wondered what was going on inside your TI-99/4a while you are sitting there with TI-Writer stoked up, your sagnum opus up there on the screen, and you are pondering what key to press next.

We'll have a look at it now from a programmer's point of view, without actually getting into the gory details of code. Instead of worrying about the sort of details that you need to know to write a loader like Funnelweb we'll take an inside out view starting right there with the blinking cursor on the screen.

This view won't go into all the possible details because the Editor is a tightly written program in assembly language and it is quite complex. So the discussion will be generally correct but will not dive off into all details along the way, or make all the caveats for particular cases.

So you are going to press a key sometime. What is the Editor doing? Waiting for you to press the key of course! It does this by sitting there in a tight little loop repeatedly calling the console keyscan routine while waiting for it to indicate that a key has been pressed.

There's more to it than that too and it's obvious to the eye - the cursor is flashing. What makes the cursor? The Editor is in text mode so it can't be a sprite and has to be another character which the key loop periodically substitutes for the character under the cursor position.

This means there has to be a counter to time the flash and maybe some extra delay to set the basic loop timing. What else is happening? If you wait long enough to make up your mind what key to press the whole screen blanks out. No big deal here because it is built into the console as a system function.

Not all home computers are so considerate of your TV. All that the key loop needs to do is enable interrupts briefly every time the key routine is executed. This is also the routine that senses the <fctn => Quit key but this has been disabled previously on the way in. The original TI-Writer Editor re-enables Quit when it goes back to GROM for SD, something I found out the hard way early in the piece.

Now you press a key. The key loop notices this and jumps out into the main key processing routine. Actually it's a little more complicated than this because it has to check the key against the last one pressed to see if it should do anything in the autorepeat line. We won't go into the details of this because writing key routines with autorepeat and flashing cursor is a devilishly tricky business.

The Editor does not use a buffer to store keystrokes. This is both a blessing and a curse. The curse is that no keys can be detected while the processor is off executing the consequences of the last keystroke. The keyboard is then dead.

Normally for text input this is over in a short enough time that it is not noticed. The blessing is that you don't suffer the drawbacks of simple minded key buffers such as multiple execution of commands if you hold the key down while waiting for some slow action to work.

First the key processor has to decide whether the key pressed is a control function of some sort as defined in the manual and if so take appropriate action. For the moment let's assume it is a plain old letter key that should put up a displayed symbol on the screen.

Hold on a moment there - where did the screen come from - how does the Editor know what to write up? It maintains in CPU RAM in low memory a 24 line set of 80 column lines, and writes up the appropriate part of this to the VDP screen display memory according to which horizontal screen window is in force and whether line numbers are to be displayed.

What is more, it does this in between every keystroke. Remember this isn't a Basic program placing characters on the screen one at a time. It isn't even the GPL/Basic line editor. This is the raw speed of the machine at work.

This sets the basic timing cycle of the Editor. If it were running on a much faster processor, say a 9995, then it would be sitting there wasting an even bigger proportion of its time once it had done all its chores after each keystroke, even for a very fast typist.

The quick screen updating by TI-Writer rather spoils your attitudes to the type of word processor that has to go out to disk for just about anything, and/or writes up the screen as a terminal.

While we are still in pursuit of the big picture, the next question to be answered is where do the CPU screen buffer contents come from. The mass of text file itself is kept in high memory along with some working buffer areas and the code for managing the buffer, about 1K in length, leaving about 23K for text buffer. How is the text stored? In the text buffer it is a little like storage of Basic programs in memory. A line number table is stored with pointers to the actual line itself. Thus whenever a line is added the line itself is merely added to the end of the buffer and only the line number table need be updated.

Deleting a line is a more serious business, as not only must the line number table be adjusted but the line itself has to be excised from the buffer and the rest of the text moved up in bulk to fill the gap. You may have noticed that deleting a range of lines is a slow process. This is because the lines are removed and the buffer adjusted one line at a time.

This is, when you think about it not a high priority area for speeding up the code or writing faster code in the first place. Faster code always takes more room. Still a smarter Delete Line would have been nice.

A more important goal is to squeeze as much text as possible into the buffer. Unlike some word processors which keep the text file on disk, only retrieving various parts of it as necessary, TI-Writer is of the type that keeps the entire document in memory.

The trade-off is between total size of document and speed of response in moving around the text. It is then important to squeeze in as large a document as possible, as the TI-99 has half of its 64K memory map assigned to various ROMs.

This is done by storing the lines in the text buffer in run length encoded form. You have heard of that in connection with all those RLE bit-mapped pictures that have been around recently.

Well, if you hadn't thought of it in connection with your TI-Writer then you are now like the character in the play who discovered he had been speaking prose all his life.

Every time a line is placed in the text buffer it is encoded, and every time a line is fetched from the buffer, whether for the screen buffer or to be written to an external device such as disk or printer, it is expanded out into its normal display form.

This extra overhead of code and processing time is the price paid for squeezing more text into the buffer. It does slow I/O to disk somewhat, and the sector interlace on disks which TI optimized for program file loading is not necessarily the fastest for TI-Writer file handling.

The difference from the picture example is that it is strings of characters that are encoded. Since lines are only 80 characters long a length count may be distinguished by setting its most significant bit, allowing a count of up to 127.

The tagged length byte is followed by the character that is to be repeated. This all means that there is no easy direct relation between the size of a text file as it is stored on disk and the amount of text buffer it occupies.

The only external indication of the state of the buffer by TI-Writer is when it is full so this is the only state you can really experiment with. So here's a little experiment.

First take a freshly initialized or swept disk to the largest capacity your drives will handle. Fill a few lines with asterisks or any other non blank character. Then use the Copy function to make as large a file of these lines as will fit. Then save the file to the empty disk.

Now back to the main story. Suppose instead of a printing character you had typed <fctn-9> or <ctrl-c> which is a lot easier on the hand. The text on the screen drops a couple of lines, and a command line appears and typed entries appear on the second line. That you know well.

A couple of minor details are interesting here. Everything you type comes out in upper case. This is done by filtering the keystrokes through a little routine that changes lower case to upper case and not by fiddling with the key-unit.

Also it doesn't matter where on the command line you start the entry because another little routine chugs along the line until it finds the first non-blank character. To this stage there haven't been great changes going on, but once a command has been decided on a major shuffle takes place.

You will recall that I said the screen buffer of 80 column lines is in low memory. Isn't that where the bulk of the Editor's code resides?

Yes it is, and when the Editor starts up it executes some initialization code and then lays down the screen buffer over this code and a whole lot more that is needed only in command mode.

That doesn't sound all that good an idea, but you have to remember that there is still a copy of that code in the machine in VDP memory.

Recall that the program file loading process as defined in the disk DSR first dumps the the program file into VDP and then it is up to the program that invoked the LOAD to do with it what it wants.

The convention with E/A program files is that the first word is a tag, null to indicate the last file, the next the length of the file, and the third the starting address where it belongs in CPU RAM.

This transfer is done nondestructively so EDITA1 is still there in VDP. So far so good but now EDITA2 is loaded and overwrites EDITA1 in VDP. No problem though because it is much shorter and doesn't overwrite the parts of EDITA1 needed later. So every time the command mode code is needed it is copied from its ghost image in VDP.

Having the command code overlayed on the screen buffer means that the visible screen image must remain frozen because the buffer is now full of code.

Equally well the program cannot be allowed to write any text into the buffer area or the code would be corrupted. When the command function is over the program refreshes the CPU buffer and then the regular loop writes the appropriate part of this to the screen and life goes on as normal. So you can see that a whole lot of shuffling is going on behind the scenes.

I can recall sorting all this code overlay process out for the first time for the first pass at what is now Funnelweb by writing a program file loader for Miniaem and then using our trusty old DIS/ASS program from Basic and printing out the disassembly in 4 columns with COLIST.

I still use that same printout for figuring out patches to the Editor. It took a whole weekend and more of poring over it, marking up the routines and branch tables.

One little confusion was that the VDP pointer in the disk file didn't seem to fit the overlay process, but was written to the correct value during program initialization. Just recently I found that was because the program was loaded with E/A during the development process and those were the values frozen in when it was saved off.

It has been noted that the DV/80 file loader is quite robust, for instance not being upset by being presented with records longer than 80 characters, merely truncating them to 80 chars.

It does this by filling a buffer area in VDP with blanks and then having the file system write the record on top of that. This way a record of a blank line saved as a single blank character is expanded out to a full line of blanks in the input buffer, and then compacted again for the text buffer.

This robustness has been compromised in Funnelweb, because at one stage the file buffers were moved up towards the pattern table during the efforts to fit in all of SD and its fully paged directory.

It turned out not to be necessary but somehow or other they never were moved back again. Issues dated Mar 87 and later will have the ORIGINAL buffer position restored.

It is not that it was ever in any danger of crashing, just that screen patterns for control characters could be corrupted by records of excess length.

At this stage I suspect that this is probably enough of a dip into the deep waters of the Editor. If it gets you thinking about how your word processor is going about its business while you are actually at the keyboard, then this article will have served its purpose, and hopefully won't have upset your typing rhythm too much.

--- Tony McGovern, Funnelweb Farm

CLOCK

PROGRAM LISTING ... in Basic

"CLOCK" by Robert T J Marshall

```

100 REM -CLOCK-
110 REM -BY ROBERT MARSHALL-
120 REM
130 REM
140 REM -DEMONSTRATION PROGRAM-
150 CALL CLEAR
160 GOSUB 1000
170 GOTO 160
980 STOP
990 REM -SUBROUTINE-
1000 T=T+.5
1010 IF T=INT(T) THEN 1030
1020 RETURN
1030 M(1)=INT(T/3600)
1040 M(2)=INT(T/60)-(M(1)*60)
1050 M(3)=T-(M(1)*3600)-(M(2)*60)
1060 FOR A=1 TO 3
1070 IF M(A)<10 THEN 1100
1080 M$(A)=STR$(M(A))
1090 GOTO 1110
1100 M$(A)="0"&STR$(M(A))
1110 NEXT A
1120 T$=M$(1)&":"&M$(2)&":"&M$(3)
1130 FOR A=1 TO 8
1140 CALL HCHAR(1,20+A,ASC(SEG$(T$,A,1)))
1150 NEXT A
1160 RETURN

```

"CLOCK" prints a digital clock in the top right hand corner of the screen and allows it to be incremented by 0.5 seconds by the use of "GOSUB 1000". Therefore, by using this statement in for-next loops and with "CALL KEY", the clock can be made to run throughout the program. The subroutine can be adapted to make the clock run faster or slower, to run backwards, to show hours and minutes only or even to display two clocks, perhaps for a two player board game.

LIFE AT "FUNNELWEB FARM"

BY MRS VAL MCGOVERN

I was asked by Steve Taylor to write a few words on my attitude to Tony's computing - on a good day. In a word - schizophrenia, swinging from being happy at seeing him deeply involved in a hobby from which he obviously gets a lot of fun and satisfaction, to considering citing the T.I. as co-respondent in a divorce action when he's still tapping away at 2am for the third or fourth night in a row.

Part of my problem of course is that I've resisted being bitten by the bug. I figure it's bad enough already having Will & Tony manoeuvring for time on the machine - the race to get up the stairs to claim the machine after dinner or even the problem of prying the possessor loose to come down to meals, without adding one more to the queue, though - sneaky thought! - if I turned into a computer addict too, they might have to cook the meal themselves to tempt me off the machine. Yes, I can see definite possibilities there.

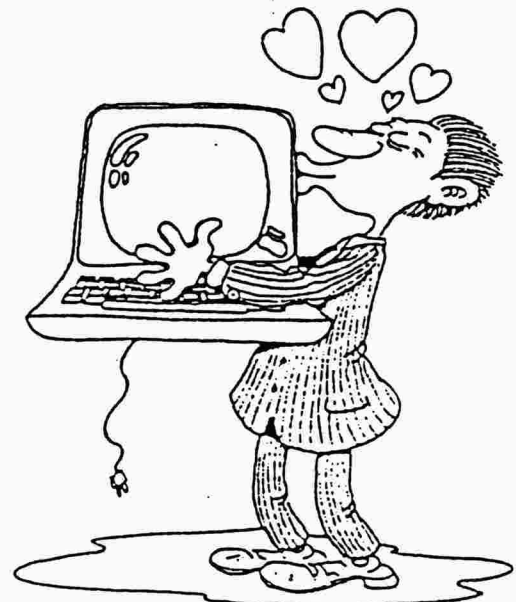
Life here at Funnelweb Farm has changed in other ways with the advent of the T.I. Friends drop by or call at all hours of the day and night, which is fun. The postbox is running hot with letters and disks from all over the world, commenting on Funnelwriter, and filling us in on T.I. happenings in their neck of the woods. All of these letters have to be answered at length with great interest, so the grass gets longer and the odd jobs get put off, but what the hell!

To get back to my reaction to computing, one of the problems which has so far discouraged me from launching myself into the world of computers is a sad lack of understanding of the fact that what constitutes fun and enjoyment to the computer devotee, may seem more like a form of masochism to us, the uninitiated.

For instance, FUN is:-

- 1) Having a program you've been working on for hours wiped out by a power fluctuation when you have forgotten to save any of it.
- 2) Accidentally wiping that particularly elegant program you'd almost finished by hitting the wrong key.
- 3) Having someone turn off the computer with your program in it when you just went out of the room for a few minutes.
- 4) Mailing off half a dozen disks of FUNNELWRITER to correspondents all over the world half an hour before your son announces with great glee that he's just found a new bug which crashes the whole thing.

Once I have convinced myself of this, and that the muttered curses, and what sounds suspiciously like shouts of agony and frustration coming from the computer room are just figments of my imagination, then move over 99'ers, here I come!



Jim Peterson

THE TEACHING COMPUTER

by Jim Peterson

I still consider the TI-99/4A to be a HOME computer, although many users have expanded it far beyond that.

And what are the uses of a home computer? Primarily, entertainment and education - and, if you can justify the expense of an RS232, printer and modem, word processing and telecommunications.

The importance of the computer in education has been overemphasized in TV advertising, to the extent that people were offended by suggestions that, if they failed to buy a home computer, they were condemning their children to a life of failure. One man even formed an organization to oppose the TI-99/4A ads!

However, the educational potential of the TI-99/4A was never realized. To teach any subject, a planned series of lessons is required. And, since each lesson is soon learned and no longer needed, the individual lessons must be inexpensive.

Texas Instruments did put out several educational modules, but they were not in any planned series and, in those days, they were not cheap. The Plato series is carefully planned, and excellently designed from an educational standpoint - but it does not take full advantage of computer capabilities, and is far too expensive.

Third party manufacturers did put out some fine educational modules but again, no planned series, and I do not know of any of them still supporting the TI.

In the early days, Micro-Ed and a few others put out some good educational software on cassette. As far as I know, Kidware and Tigercub are the only ones still offering educational programs on cassette.

There is also a vast amount of public domain software written by amateur programmers. It ranges in quality from mediocre to excellent, consists mostly of pre-school teaching or basic math drills, and would take much effort to organize into any course of education.

The market for educational software has been so poor that it is doubtful that any more will be produced. So, if you need it, you'll have to write it for yourself!

Ideally, educational programs should be written by teachers, because they

know how to teach. Unfortunately, it seems that few teachers have learned how to write programs, although some of their students have. A cooperative effort could have resulted in a large pool of good educational public domain software available to schools. However, the educational establishment as a whole has been so brainwashed by the Apple peddlers and the disciples of Logo that no support is given to anything else.

What makes a good educational program? If possible, it should be interactive - it should, 1. teach a lesson; 2. test to see if you have learned the lesson; 3. if you have not, go back and teach it again; 4. if you have learned, go on to the next lesson.

This is not always practical in a single program, but can be done by having one program run the next program. I used a variation of this technique in "Casting Out Nines", which teaches a method of checking long multiplication and division problems. The student must correctly solve several problems at each step before the program will continue to the next step.

The program should give some kind of spoken, printed, graphics or musical reward for correct answers, and a corresponding admonition for wrong answers. The nature of these depends on the age level of the student. In "Kindermath" I use a groan and a frowning face which can be changed to a smile and music by a correct answer. In "Kinderminus", the problem is displayed in the center of a multicolored kaleidoscope which changes patterns for each correct answer or turns black for a wrong answer. For an older student, a short game to be played could be offered after successfully answering a series of questions.

Computer graphics and sound should be fully utilized, but not allowed to become boring. The same musical salute after each correct answer soon gets tiresome. In some of my programs, a note is added to a tune for each correct answer, and after several such answers the entire tune is played. "Kindermath" uses several nursery tunes in succession.

The "stupid computer syndrome" should be avoided. This occurs when random selection causes the same question to be asked twice in succession. It is easily prevented by the simple statement IF

Q=Q2 THEN (go back for another selection) ELSE Q2=Q.

The computer is especially well adapted to teaching math, because it can generate an infinite number of random problems to be solved. When the problem requires keying in an answer to a multi-digit problem displayed on the screen, input should be accepted from right to left in the same way as it was being worked on paper. My "Math Homework Helper" does this with addition, subtraction and multiplication problems of any size, and also helps the student by refusing to accept an incorrect digit.

When possible, if a wrong answer is given the student should be shown how to work the problem. In one of our Extended Basic classes we analyzed a math quiz program which generated random problems in the form of "IF 3 BOYS CAN CATCH 12 FROGS IN 4 DAYS, HOW MANY FROGS CAN 9 BOYS CATCH IN 8 DAYS?". If the answer was wrong, a screen display explained, "NO, THAT'S WRONG. IF 3 BOYS CAN CATCH 12 FROGS IN 4 DAYS, THEN 3 BOYS CAN CATCH 3 FROGS IN ONE DAY".....etc., through the problem. Similar routines could be written for a wide variety of time/speed/distance problems, etc.

The basics of music education can also be easily taught by computer. The TI-99/4A can generate any musical tone required, and the piano keyboard, guitar fingering, musical notation, or whatever can be graphically displayed. John and Norma Clulow, Regena, and Bob Pomictter have written some excellent programs of this type.

Educational programs requiring much text are more difficult because of the limited memory capacity of the computer. Many of the public domain programs of this kind seem to have been typed in directly from a textbook, and there is really little reason for computerizing them. Some good module software is also of this type - nicely programmed, but very soon learned and discarded.

Most speed reading programs on the market are loaded with a data base of sentences, which are flashed on the screen briefly and the student is then asked to repeat them. He soon begins to recognize them from memory, even though he may think he is reading faster. Also, the purpose of speed reading is to grasp the meaning of a sentence, not its

exact wording. My "Speeder Reader" and "Junior Speeder Reader" programs bypass these faults by being loaded with a wide selection of individual nouns, verbs, adjectives and modifiers which are randomly selected and combined into an infinite number of sentences. Then, the student is asked any one of several different questions about the sentence, requiring a one-word reply.

Public domain spelling programs are popular but not very practical. In order to tell the student to spell a word, it must be either printed on the screen or spoken by the speech synthesizer. In the former case, he is already being shown how to spell it. The vocabulary of the speech synthesizer is rather small. The TE II permits an unlimited vocabulary, but the pronunciation is not very clear - a sentence may be understood, but a single word out of context is more difficult. Ron Binkowski published a "Speller" program in the 99'er which overcame this problem by allowing a separate phonetic spelling of the word and an "as in" phrase which could be phonetically spelled until it sounded right.

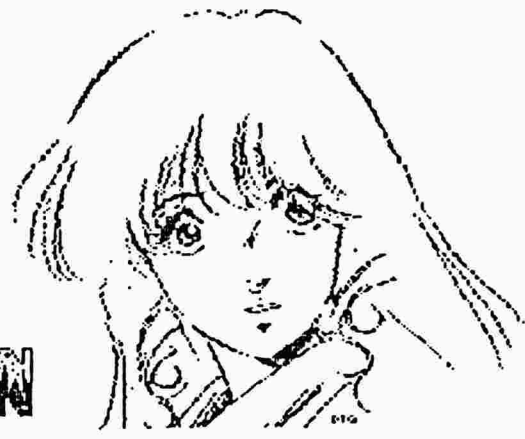
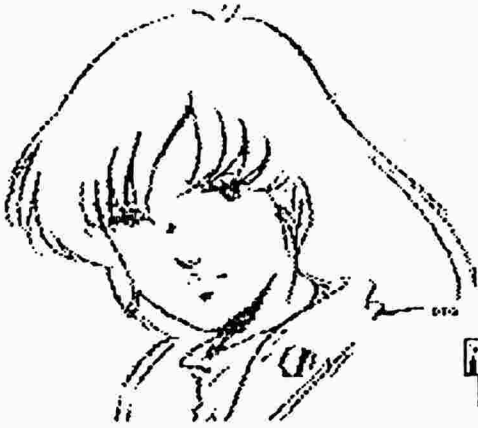
In my "Miss Spell", I programmed each word in a correct and incorrect spelling, randomly showed either one to the student and asked him if it was correct and, if not, to spell it correctly. In "I & E Spelling" I programmed every word containing those difficult "ie" and "ei" combinations, randomly showed them on the screen with those two letters replaced by blanks, to be filled in. If the answer is wrong, the student is shown a screen displaying the "I BEFORE E EXCEPT AFTER C" rule with all its exceptions.

Finally, the best educational programs of them all are those that teach a person something while he thinks he is just having fun. Word games are of this type, whether the popular "Hangman" or "Scrabble" or whatever.

My "Tirkle" is just a very simple little game for children, based on the early computer game called "Hurkle". However, teachers have told me that they find it very useful for teaching young children logical thinking and compass directions.

The possibilities are endless - and so little is being done!

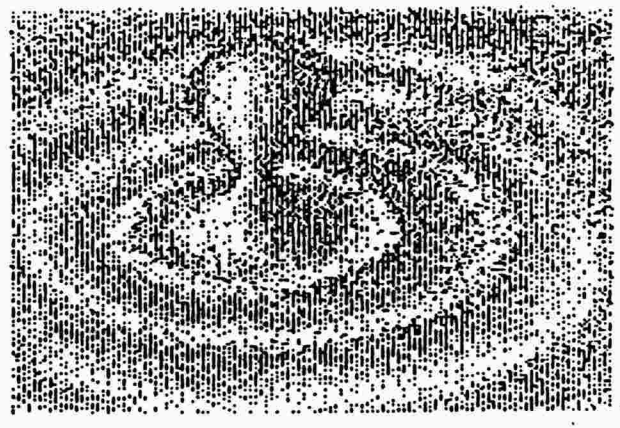
SARSAPARILLA SAL



THEN

Once I was an ordinary schoolgirl...

I tasted SARSAPARILLA...

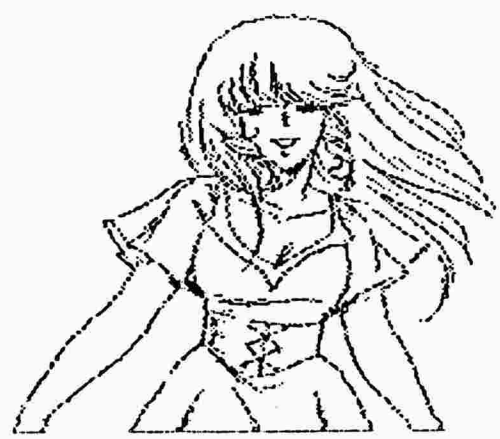


it was another world...
my life changed...
I met interesting people...

Printed in one pass using
The Printers Apprentice
Put together by
Stephen Shaw



I was never the same again...



* - FREE AD - *

HUNTERVALLEY99'ERSHUNTERVALLEY99'ERSHUNTERVALLEY99'ERSHUNTERVALLEY99'ERSHUNTE
HUNTERVALLEY99'ERSHUNTERVALLEY99'ERSHUNTERVALLEY99'ERSHUNTERVALLEY99'ERSHUNTERV

MEMBERSHIP - HUNTER VALLEY 99'ERS
1987 / '88

Membership subscriptions for the year from the 1st JULY 1987
to 30th JUNE 1988
are NOW DUE and should be paid prior to the 30th JUNE 1987

Please complete the following membership renewal form and forward
the FORM and PAYMENT to :-

The Secretary
HUNTER VALLEY 99'ERS
6 Arcot Close,
TARRO - NSW. - 2322
AUSTRALIA

NAME :- _____
(in full)

ADDRESS :- _____
(in full)

PHONE No >>(H) _____

SUBURB :- _____

if you like>(W) _____

TOWN :- _____

XXXXXXXXXXXXXXXXXXXX

STATE :- _____

* PLEASE PRINT *

COUNTRY :- _____

* CLEARLY *

ZIP or POST CODE :- _____

XXXXXXXXXXXXXXXXXXXX

Please find enclosed a CHEQUE (etc.) _____ for \$40:00(AUS) being
payment for my ANNUAL MEMBERSHIP FEES for the HUNTER VALLEY 99'ERS
for the year 1st JULY 1987 to 30th JUNE 1988. (** Note :- \$30:00
overseas surface)

I agree to abide by the Rules and Constitution of the
Hunter Valley 99'ers.

Signed :- _____

Dated :- _____

HUNTERVALLEY99'ERSHUNTERVALLEY99'ERSHUNTERVALLEY99'ERSHUNTERVALLEY99'ERSHUNTE
HUNTERVALLEY99'ERSHUNTERVALLEY99'ERSHUNTERVALLEY99'ERSHUNTERVALLEY99'ERSHUNTE

* Overseas Surface = A\$ 30
(cost about UK£14)

MONTHLY A4 SIZE
NEWSLETTER - About 44 pages
- Plenty on Forth! -

CASSETTE LIBRARY UPDATE

As promised, our cassette librarian, Maurice Rymill, has added his personal collection of tapes to the library. These were collected in the good old days when some of the 'house' magazines supported the TI. Maurice will transfer any of these programs to tape for £1.25 per program, plus the cost of the tape. Or he would put 4 programs on one tape at £1 per program. So you could have 4 programs for £5.50 (which would include cost of tape and post and packing). Maurice has done some reviews, which I'll put in, space permitting.

No.	TITLE	DESCRIPTION	REFERENCE	No.	TITLE	DESCRIPTION	REFERENCE
1	"BUG" MAKER	EXBAS EXP	E2 B9	45	CHAMELEON	BAS UTIL	E3 B7
2	AIR RAID	BAS GAME	G8 B3	46	CHANNEL PATROL	BAS GAME	G15 B2
3	AIRSHIP RESCUE	EXBAS GAME	G22 A4	47	CHASE	BAS GAME	U1 B4
4	AIRSTRIKE	BAS GAME	G9 B2	48	CHOROPLETH MAPS	EXBAS UTIL	U19 B1
5	ALIEN ATTACK	BAS GAME	G10 B2	49	CHUCK-A-LUCK	EXBAS GAME	G36 A1
6	ALIEN INVADER	BAS GAME	G4 A4	50	CHURCH BELLS	EXBAS MUS	M1 A4
7	ALIEN INVADER 1	BAS GAME	G10 A4	51	CJRC. + POINTPLOT	BAS UTIL	E3 A1
8	ALIEN INVADER 2	BAS GAME	G10 A5	52	CIRCUS	EXBAS GAME	G34 A1
9	ALONE AT SEA	BAS ADU.	G5 A1	53	CODEBREAKER	EXBAS GAME	G12 B3
10	ALPHA T'OMEGA	EXBAS GAME	G27 B1	54	COMP-AIDED DESN	BAS UTIL	U11 A2
11	APPLE SCRUMP	BAS GAME	G15 A1	55	COMP. VISUALS	BAS UTIL	U11 A4
12	ASTEROID RESCUE	EXBAS GAME	G24 A4	56	COMPUTER CARD	EXBAS GAME	G37 A1
13	ASTRO GALLERY	BAS GAME	G8 B4	57	CONNECT 4	BAS GAME	G9 B3
14	BAND AID	BAS MUS	S1 A2	58	COSMIC DIGGER	EXBAS GAME	G27 B3
15	BANZAI BUNNY	EXBAS GAME	G14 B3	59	CRAZY 'COPTER	BAS GAME	G9 B1
16	BAR GRAPH	BAS UTIL	E3 B6	60	CUBIK'S RUBE	BAS GAME	G8 B2
17	BARREL ORGAN	BAS MUS	E1 A5	61	DAM RAIDERS	BAS GAME	G26 A1
18	BARS + GRAPHS	BAS UTIL	U5 A3	62	DATA FILER	EXBAS UTIL	U8 A1
19	BASEMENT BOB	EXBAS GAME	G36 B1	63	DBL HGHT CHSET	BAS UTIL	E3 B8
20	BATTLE AT SEA	BAS GAME	G3 B1	64	DEATH RACE	BAS GAME	G16 B3
21	BATTLESHIPS	BAS GAME	G34 B2	65	DEBROIDS	EXBAS GAME	G19 B2
22	BEAGLE HIKE	EXBAS GAME	G30 B4	66	DEC-HEX :HEX-DEC	EXBAS EXP	E1 A8
23	BEE-LINE	EXBAS GAME	G23 B1	67	DEFUSE	BAS GAME	G10 B1
24	BEETHOVEN CLASS	EXBAS MUS	G29 A4	68	DIAMOND MINER	BAS GAME	G6 B1
25	BEGIN. BAS. TUTOR	BAS EDUC	G9 A1	69	DJVE BOMB	BAS GAME	G18 A1
26	BJORHYTHM	EXBAS UTIL	G19 A1	70	DOG FIGHT	EXBAS GAME	G35 B1
27	BJORHYTHMS	BAS UTIL	G20 B3	71	DRAWING BOARD	BAS UTIL	U2 B2
28	BLACK BOX	BAS GAME	G3 B2	72	EARTH DEFENCE	BAS GAME	G7 B3
29	BLACKJACK	BAS GAME	G4 B1	73	EAT MINCE PIES	BAS GAME	G9 B21
30	BOA ALLEY	BAS GAME	G23 B3	74	ELECTRON	BAS GAME	G9 B13
31	BOMBER 1	BAS GAME	G4 B2	75	ELECTRON GRID	BAS GAME	G1 A1
32	BOMBER 3	BAS GAME	G6 B3	76	ERIC ADDS UP	BAS EDUC	G1 B4
33	BOMBS AWAY	BAS GAME	G15 B4	77	ESMERALDA	BAS EXP	E3 B4
34	BOWLS	BAS GAME	G2 B3	78	EVASION	BAS GAME	U1 B2
35	BREAKOUT	BAS GAME	G5 B1	79	EXBAS TUTOR	EXBAS EDUC	U13 A1
36	C/PILLAR CRAWL	BAS GAME	G18 B1	80	FACE IN SPRITES	EXBAS UTIL	U11 B1
37	CALENDAR 1	BAS UTIL	E3 B2	81	FAMILY TREE	EXBAS UTIL	G34 A2
38	CALENDAR 2	BAS UTIL	P2 A1	82	FEM ON THE GRID	BAS GAME	G14 A2
39	CANNIBALS	EXBAS GAME	G34 B3	83	FINANC. WIZARD	EXBAS UTIL	E2 B7
40	CARD SHARK	BAS GAME	G16 B2	84	FIREMAN	BAS GAME	G3 A2
41	CARDS	BAS EXP	E2 A1	85	FLYING SAUCERS	BAS GAME	G16 B1
42	CASSFILE SYSTEM	BAS UTIL	U12 A1	86	FOLLOW THIS	EXBAS GAME	G35 A2
43	CATERPILLAR	BAS GAME	G6 B6	87	FOOD HWNT	BAS GAME	G14 A1
44	CAVERN ATTACKER	BAS GAME	G15 A2	88	FOREST RALLY	BAS GAME	G7 B2

CASSETTE LIBRARY UPDATE

No.	TITLE	DESCRIPTION	REFERENCE	No.	TITLE	DESCRIPTION	REFERENCE
89	FORKLIFT TRUCK	BAS GAME	G32 A2	1	"PHONE"(REPORT)	PRG UTIL	U9 B1
90	FORMULA 1	EXBAS GAME	M4 A2	2	"STORE""RECALL"	MMM UTIL	U8 B4
91	FREEWAY 2000	EXBAS GAME	G36 B2	3	MAGIC SQUARE	BAS GAME	G5 A2
92	FROGGER	EXBAS GAME	G12 B5	4	MEMBERS FILE(P)	EXBAS UTIL	P1 A4
93	FROGLING	EXBAS GAME	G27 A1	5	MEMORY MAZE	BAS GAME	G6 B8
94	FRUIT COCKTAIL	BAS GAME	G8 A4	6	MEMORY RELOCATR	MMM UTIL	U23 A1
95	FRUIT EAT SNAKE	BAS GAME	G42 A1	7	METEOR ATTACK	EXBAS GAME	G30 B1
96	FRUIT MACHINE	BAS GAME	G4 A2	8	METEOR ATTACK	BAS GAME	G8 A2
97	FUR ELISE	EXBAS MUS	G14 B1	9	METEOR STORM	BAS GAME	G2 A1
98	GAME OF CHASE	BAS GAME	G26 A6	10	METEOR STORM 2	BAS GAME	G26 A5
99	GIANTS + DWARFS	EXBAS GAME	G23 B4	11	MICRO CHECKERS	BAS GAME	G5 B3
100	GLOBAL RESCUE	EXBAS GAME	G13 B1	12	MINE MAZE	BAS GAME	G6 A1
101	GOLF 2	BAS GAME	G11 A2	13	MINI-MEM DEMO	MMM UTIL	U3 A1
102	GORILLASSI	BAS GAME	G6 A2	14	MISS MUFFET	BAS GAME	G6 B2
103	GRAPHICS MAKER	BAS UTIL	U10 A1	15	MISSILE DUEL	EXBAS GAME	G33 B3
104	GREENSLEEVES	BAS MUS	M1 A3	16	MMM SAMPLE DATA	MMM UTIL	G32 B1
105	GUNFIGHT	BAS GAME	G22 A3	17	MOOS	EXBAS GAME	G22 B3
106	H/BACK RESCUE	EXBAS GAME	G22 A5	18	MOSAIC	MMM DEMO	U8 B2
107	HALLEY'S COMET	EXBAS GAME	G35 A3	19	MOTH MANIA	EXBAS GAME	G22 B2
108	HANG GLIDER P1L	EXBAS GAME	G13 A2	20	MOTORWAY	EXBAS GAME	G23 B2
109	HANGMAN	BAS GAME	G31 B3	21	MOVING STARS	EXBAS DEMO	G15 B2
110	HAUNTED HOUSE	BAS ADV	G12 A3	22	MOZART	BAS MUS	M1 A5
111	HJ-RES X/Y PLOT	BAS EXP	E3 B5	23	MULTI-COLORS	BAS DEMO	E2 B6
112	HJRES EDITOR	BAS EXP	E3 B3	24	MULTIC	MMM DEMO	U8 B3
113	HOME SECRETARY	BAS UTIL	U2 A3	25	MUNCHER	BAS GAME	G6 A3
114	HORSE RACE	BAS GAME	G3 A1	26	MUSIC CONTRDL	EXBAS MUS	G29 A1
115	ICON SIMULATION	EXBAS EXP	E4 A2	27	MUSICAL COLORS	BAS MUS	G20 B1
116	INVADER	BAS GAME	U1 B1	28	NAME THAT BONE	BAS EDUC	U11 A3
117	INVENTION NO. 13	EXBAS MUS	M1 B2	29	NIGHT BIRDS	EXBAS DEMO	G30 A2
118	JEWEL MINING	BAS GAME	G1 A2	30	NUCLEAR RACE	BAS GAME	G8 B5
119	JUNGLE FEVER	EXBAS GAME	G19 B4	31	ORGAN	BAS UTIL	U1 B3
120	JUNGLE JIM	EXBAS GAME	G38 A3	32	OTHELLO II	BAS GAME	G17 A4
121	KIPPY'S NGHTMRE	MMM GAME		33	OTHELLO III	BAS GAME	G16 B5
122	KITTEN KONG	EXBAS GAME	G19 B3	34	PAPERHANGER	BAS MUS	G26 B3
123	KRAZY KIDD	BAS GAME	G18 A4	35	PARATROOPER	EXBAS GAME	G24 A3
124	L-B-L ASSEMBLER	MMM UTIL	U15 ALL	36	PEARL DIVERS	BAS GAME	G15 A4
125	LABELS	EXBAS PRIN	P1 A2	37	PENGUIN	BAS GAME	G6 B4
126	LADDERS+ADDERS	BAS GAME	G8 A1	38	PENTATHLON	EXBAS GAME	G13 A1
127	LASER BIKE RACE	BAS GAME	G9 B4	39	P1LOT	BAS GAME	G7 B4
128	LASER DEFENCE	BAS GAME	G8 A3	40	PINK PANTHER	BAS DEMO	E3 B1
129	LAST STAND	BAS GAME	G2 A3	41	PLANE	EXBAS DEMO	G30 A1
130	LEAPFROG	EXBAS GAME	G22 B1	42	PLANET TREK	BAS GAME	G40 B1
131	LIFT ATTENDANT	BAS GAME	G9 B16	43	PLAY CARD RIGHT	BAS GAME	G16 B2
132	LINES	MMM DEMO	G30 A3	44	PLAYING/PIXELS	BAS UTIL	U1 A1
133	LOST RUINS	BAS GAME	G2 B2	45	POCKET INVENT	BAS UTIL	G2 B2
134	LOWER CASE LETS	BAS EXP	E1 A3	46	POMPEII	EXBAS GAME	G12 B2
135	LUNAR LANDER	EXBAS GAME	G21 B2	47	POOLS SELECTOR	BAS UTIL	E2 B5
136	LUNAR LANDING	BAS GAME	G9 A10	48	POTHOLE POTTY	BAS GAME	G4 A5

CASSETTE LIBRARY UPDATE

No.	TITLE	DESCRIPTION	REFERENCE	No.	TITLE	DESCRIPTION	REFERENCE
49	POWER BOAT	BAS GAME	G3 A3	99	SUBSTRIKE	EXBAS GAME	G12 B4
50	PRELUDE(BACH)	EXBAS MUS	G26 B4	100	SUICIDE RUN	BAS GAME	G29 B1
51	PRINT COL GRAPH	BAS UTIL	P1 A1	101	TANK BATTLE	BAS GAME	G9 B14
52	PRISON RUN	BAS GAME	G26 A3	102	TARGET	EXBAS GAME	G12 B1
53	REACT	BAS GAME	G9 B8	103	TCH SELF EXBAS	EXBAS EDUC	U13 A+B
54	RESCUE	EXBAS GAME	G38 A1	104	TCH Y/SELF BASC	BAS EDUC	U14 A+B
55	ROAD RACE	BAS GAME	G2 A2	105	TEX INVAD REVEN	BAS GAME	G8 B1
56	ROAD RACE	MMM GAME	G32 B1	106	TEX TELEPHONES	BAS GAME	G17 A3
57	ROAD RACER	BAS GAME	G15 A3	107	THE PUZZLER	BAS GAME	G40 A1
58	ROAD RACING	BAS GAME	G42 B1	108	THREE-D	BAS EXP	E2 B2
59	ROADER	BAS GAME	G29 B2	109	THREE-D MAZE	BAS GAME	G6 A4
60	ROBOT FIREWATCH	BAS GAME	G7 A2	110	THREE-D PICTURE	BAS UTIL	U1 A4
61	ROLLERGOAL	EXBAS GAME	G32 B2	111	THREE-D SHAPES	BAS EXP	E1 A6
62	RUNNER/TREITON	BAS GAME	G37 B1	112	THREE-D SPRITES	EXBAS EXP	E2 A7
63	SAMANTHA I + II	BAS EXP	E4 B2	113	T1 SOUND SHAPER	BAS MUS	M1 A1
64	SCOREBOARD	BAS UTIL	E1 A1	114	T1 SYMBOL	EXBAS EXP	E2 B1
65	SCRABBLE BOARD	EXBAS EXP	E1 B1	115	TINY PLAN	EXBAS UTIL	U5 A2
66	SCREEN DUMP	MMM UTIL	U8 B5	116	TOWER OF HANOI	BAS GAME	G3 B3
67	SEA DIVER	EXBAS GAME	G27 B2	117	TOWERS OF BRACO	BAS GAME	G4 B4
68	SEA WOLF	BAS GAME	G32 A1	118	TRAFFIC COP	EXBAS GAME	G35 B3
69	SHEEPDOG TRIALS	BAS GAME	G9 B12	119	TREASURE ISLAND	BAS GAME	G1 B2
70	SHOEBOX(FILING)	BAS UTIL	U1 B7	120	TRENCH ATTACK	EXBAS GAME	G41 A1
71	SIMON	BAS GAME	G6 B5	121	TURN OF CARD	BAS GAME	G32 A3
72	SINGLE PIX DRAW	EXBAS UTIL	E1 B3	122	TURTLE HOP	BAS GAME	G7 A1
73	SIR PRANCE-ALOT	EXBAS GAME	G19 B1	123	TYPWRITER	EXBAS UTIL	U6 A1
74	SKEET SHOOT	EXBAS GAME	G36 A2	124	UNION FLAG TUTR	EXBAS EDUC	U21 B1
75	SKI-RUN	BAS GAME	G3 A4	125	VENETIAN BOATS	EXBAS MUS	M1 B1
76	SKIING	BAS GAME	G4 B3	126	VERBOSE(SPEECH)	EXBAS UTIL	U7 B1
77	SKITTLES	BAS GAME	G26 A2	127	VIDEO CREDITS	EXBAS UTIL	M2 A3
78	SLALOM	BAS GAME	G7 A5	128	VIDEO TITLES	EXBAS UTIL	E2 A6
79	SLUGS + LADDERS	BAS GAME	G7 A4	129	VIDEO TITLING	EXBAS UTIL	E1 B4
80	SLUGS U SNAJLS	BAS GAME	G7 A3	130	W/MINSTER CHIME	EXBAS MUS	G20 B4
81	SNAKE	BAS GAME	G6 B7	131	WALKING SPRITES	EXBAS DEMO	E2 A4
82	SNAP-CALC	EXBAS UTIL	U2 A3	132	WEEDKILLER	BAS GAME	G26 A4
83	SNOWDROP	BAS GAME	G9 B21	133	WHEEL (SPRITE)	EXBAS DEMO	E2 A3
84	SONAR	EXBAS GAME	G29 A2	134	WILD KINGDOM	EXBAS GAME	G33 A1
85	SORCER'S CASTLE	BAS ADV	G9 B9	135	WORD MAKER(SP)	EXBAS UTIL	U7 A1
86	SORTING	EXBAS UTIL	U5 B1	136	WORD PROC. 1	BAS UTIL	U4 A1
87	SOUND ADVICE	EXBAS MUS	M2 A4	137	WORD PROC. 11	BAS UTIL	U4 A2
88	SOUND TUTORIAL	BAS MUS	S1 A1	138	WORD PROCESSOR	EXBAS UTIL	G29 A3
89	SPACE LASER	BAS GAME	G1 B1	139	WORD SEARCH	BAS GAME	G18 A2
90	SPEEDRACE	EXBAS GAME	G24 B1	140	WORD WORLD	EXBAS GAME	G24 A1
91	SPRITE EDITOR	EXBAS UTIL	E1 B2	141	WORDPRO(PRINT)	EXBAS UTIL	U5 A1
92	SPRITE SHADOWS	EXBAS EXP	E2 A8	142	WORDSORT	BAS EXP	E1 A2
93	SPRITES IN 3-D	EXBAS EXP	E2 A7	143	XMAS CARD	BAS UTIL	U19 B2
94	SQUADRON LEADER	BAS GAME	G29 B3	144	YACHT RACE	BAS GAME	G16 B4
95	STAR DUEL	BAS GAME	G18 A3	145	ZONE DEFENDER	BAS GAME	G14 B2
96	STAR PATROL	BAS GAME	G5 A3				
97	STARS	EXBAS EXP	E2 A2				
98	STARZONE FIGHT.	BAS GAME	G11 A1				

Conditions of the Group Library

To receive a free program on cassette/disk we ask members to submit their own programs for the Users-group software library. Programs must be original and NOT copy from commercial software houses or third party copyright programs (unless written consent is obtained to use such programs). Instructions on use should always be included.

When a member submits a program to TI99/4a Users Group library it will remain the property of the original author who either coded or translated the program. TI99/4a Users Group UK does not claim any proprietary rights to any program listed in the software exchange library and cannot be held responsible for their contents.

All programs will be made available exclusively to members. This group will have the right to use such software in exchange of other users programs. IT IS MOST IMPORTANT that members using the software exchange library comply with the COPYRIGHT LAW. Anyone breaking this rule will face legal action being taken. None of the programs can be sold commercially.

If you have a program to exchange send in your tape or disk and we will return a free program of your choice. IMPORTANT please enclose envelope and stamps for return. Members who do not have a program to donate can ORDER for a handling fee of £4.50 any THREE programs (THIS MEANS YOU CAN KEEP THE PROGRAMS FOR YOUR OWN USE). Additional programs are charged at £1.75 each, or £4.50 for multiples of three. This fee includes cost of tape handling, postage, packing (DISK ADD 95p). It should be noted that overseas orders will be sent surface mail. U.K. FUNDS ONLY. CHEQUES crossed made payable to TI99/4a Users Group UK.

****NOTE**** We cannot accept orders from NON MEMBERS.

MEMBERS APPLICATION FOR SOFTWARE.

I _____ (PRINT NAME) agree as a member with the rules of the Group library relating to Copyright.

signed _____ date _____ (file _____ code _____ date _____ fee _____) (office use only)

I wish to Exchange/order:-(1) _____ (2) _____ (3) _____

PROGRAMS ON CASSETTE _____ DISK _____ (add 95p) ADDITIONAL PROGRAMS

PLEASE LIST SEPARATELY

This is your address label please print clearly

NAME _____
ADDRESS _____

PLEASE SEND YOUR CHEQUE OR PROGRAM WITH THIS SIGNED APPLICATION FORM TO:-
TI99/4a Users Group (UK)
GROUP LIBRARY
231, BOURNVILLE LANE,
BOURNVILLE,
BIRMINGHAM
B30 1RA U.K.

APPLE SCRUMPING - BAS GAME G15 A1

Move around an orchard, picking up apples, avoiding the farmer and his baskets, to find a key to the orchard gate. Set your own time limit in seconds.

CHANNEL PATROL - BAS GAME G15 B2

Attempt to torpedo enemy shipping, while avoiding being rammed or depth-charged.

CAVERN ATTACKER - BAS GAME G15 A2

Move your craft through a cavern, attempting to avoid stalactites, while also trying to avoid enemy lasers, to bomb the enemy nerve centre.

FOREST RALLY - BAS GAME G7 B2

Drive your car through the forest from start to finish, avoiding the trees, to keep a low score.

EARTH DEFENCE - BAS GAME G7 B3

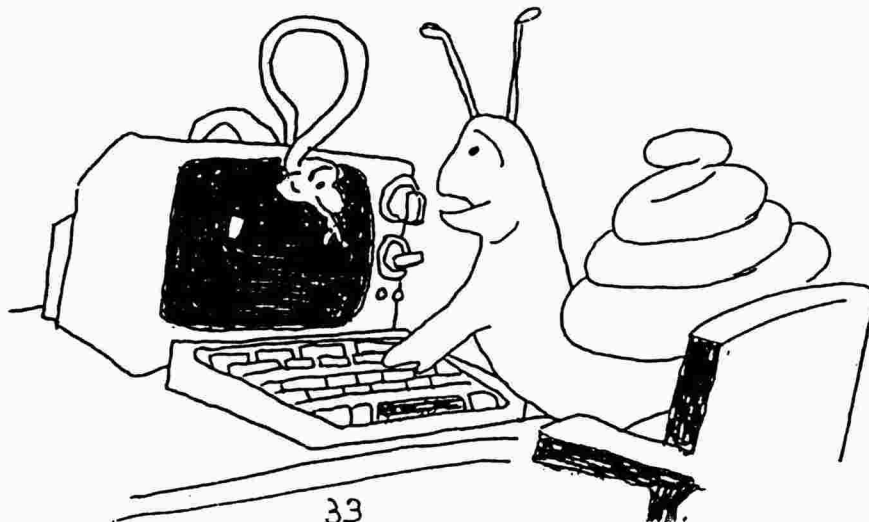
Shoot down the aliens using your ack-ack gun, before your time or ammunition runs out.

SLUGS V SNAILS - BAS GAME G7 A3

Two players each use half of the keyboard, in turn, to eat as many lettuce as possible before their opponent does the same.

SLUGS AND LADDERS - BAS GAME G7 A4

Use your skill to kill all the slugs in the garden, before they get you, or your oxygen runs out. Several skill levels.



BUGS IN R. MOLESWORTH BOOK
"INTRODUCTION TO ASSEMBLY
LANGUAGE FOR THE TI HOME
COMPUTER"

By Rob Williams TIUP 1/9/1984

Page 49
Replace: VDP RAM ADDRESS DECIMAL=
((ROW-1)*32)+(COLUMN-1)
with: VDP RAM ADDRESS DECIMAL=
((ROW-1)*32)+(COLUMN-1)

Page 67
Replace the 11th line of the
program reading: C R7,4
with: CI R7,4

Page 68
Replace the 4th line of the
program reading: CI R3,R7
with: C R3,R7

Page 74
Replace the 16th line of the
program reading: BYTE >0F
with: BYTE >05
in order to make the program to
branch to the Main Title Screen
when Quit is pressed.

Page 76
Replace the first line of the
program reading: AORG >7D0C
with: AORG >7D30
This is because enough space has
not been allocated in the book
for the Symbol Table. For
further information read page 36
of the book.

Replace the 15th line of the
program reading: DATA >0F00
with: DATA >0500
in order to make the program to
branch to the Main Title Screen
when Quit is pressed.

Replace the 5th line from the
bottom of the page: BLWP @KS N
with: BLWP @KS

Page 89
Replace the 13th line of the
program reading: JLE CPUT
with: JL CPUT
Otherwise 11 records will be
written, not 10.

Page 118
Replace the 8th line of the
program reading: CLR STATUS
with: CLR @STATUS

Page 119
Replace the 2nd line from the
page bottom reading: LI R0,0755
with: LI R0,>0755

Page 121
Insert after line 4 these two
lines: CB @KEYVAL,@ESCV
JEQ ESCAP

Page 122
Replace the 10th line of the
program: 100 FOR LOOP=0 TO 16
with: 100 FOR LOOP=0 TO 14

Page 123
Replace the 4th line of the
program reading: SBUFF BSS 256
with: SBUF BSS 256

Page 124
Replace the 25th line of the
program reading:
ERROR LI 0,>1300
with: ERROR LI R0,>1300

Replace the 26th line of the
program reading: BLWP @ER R
with: BLWP @ERR

Page 127
Delete three words from the 7th
line of text: "or equal to"

Page 130
Add one new line after line 20,
ie between char 90 and 91:
DATA >0000,>0000

I have discovered that 4 bytes
of zero's must be added here to
force correct loading in the
Pattern Description Table of all
characters from 91 to 143.

Page 131
Replace the 4th line from the
page bottom reading: LI R2,904
with: LI R2,912

TIUP



FROM TIT-BITS VOL 3 NO 2
TI USERS PERTH (Australia)

ANIMATED SPRITES

DEMONSTRATION

This is a demonstrartion on how sprites can be animated.

FROM TIT-BITS VOL 4 NO 3 FEB 1986

--TI Users PERTH (Australia)--

100 REM

FLYING HORSE SPRITES

110 REM

LA 99ERS

(TOPICS SEPT 1985)

(VIA A9CUG CALL NEWSLETT

ER SAVANNA 99ERS)

120 CALL CLEAR

130 CALL CHAR(96,"0000041B05

3B01314F8F8F1C30282410000000

88B86C62E0E0E0F82424488000")

140 CALL CHAR(100,"03000C020

D0321514F8F0F3C5060A01000804

088BC6864E0E0E0F82412090400"

)

150 CALL MAGNIFY(4):: CALL S

PRITE(#1,96,2,90,90,#2,100,2

,90,90)

160 CALL MOTION(#1,-1,18,#2,

-1,18)

170 FOR FLY=1 TO 40 :: FOR A

=1 TO 2

180 CALL COLOR(#A,1):: CALL

COLOR(#(3-A),2)

190 FOR X=1 TO 90 :: NEXT X

:: NEXT A :: NEXT FLY

200 CALL MAGNIFY(3):: GOTO 1

70

Hmm...

Member Phil Trotter went to market one day and stopped by a stall selling used cassettes, records and video tapes. What is this? A Stainless Software cassette? Says stall holder..."I don't know what it is but the music's RUBBISH, it's yours for 50p". DONE! Said cassette was OUTLAND by Roach Software, which sold in the USA for US\$16. When Phil got it home it loaded first time! Keep your eye on Market stalls!

REQUEST. Please.

Unfortunately my father is fast losing his sight and wants to learn touch typing. I can't help but feel that a TI plus TE2 plus Speech Synth could come in handy. Can anyone help me get the equipment together without going overboard on cost??? I can find a spare console if required, but the TE2 and speech synth seem hard to get these days... cassette leads would be nice too! Please contact Stephen Shaw, 10 Alstone Road, STOCKPORT, Ches, SK4 5AH. Or telephone, between 8pm and 10pm, 061-432 6097.

Try this short program for the effects it can create. Don't be surprised at what happens. It will run in either basic.

FROM TIT-BITS VOL 4 NO 3 FEB 1986

--TI Users PERTH (Australia)--

100 DIM E(24,32)

110 FOR A=32 TO 127

120 CALL CHAR(A,A\$)

130 B\$=""

140 FOR B=15 TO 1 STEP -2

150 C\$=SEG\$(A\$,B,2)

160 B\$=B\$&C\$

170 NEXT B

180 CALL CHAR(A,B\$)

190 NEXT A

200 FOR C=1 TO 24

210 FOR D=1 TO 32

220 CALL GCHAR(C,D,E(C,D))

230 NEXT D

240 NEXT C

250 FOR F=1 TO 24

260 FOR G=1 TO 32

270 CALL HCHAR(25-F,G,E(F,G))

)

280 NEXT G

290 NEXT F

300 GOTO 110

From THE MSP99 NEWSLETTER Vol 3 No 6

JUNE 1984

MSP 99 USERS GROUP, ST PAUL, MINNESOTA

To use a parallel printer with the Weight Control and Nutrition Module, when asked at the beginning of the program "WILL YOU BE USING A PRINTER? Y/N" answer "N". Later in the program, on one of the menus, you have the option to print records. Upon making this selection you are asked to enter the device name. This time enter "PIO". You will be given the choice of printing either Personal Information or Menus. (From the Ozark 99'ers Users' Group.)

DIAGNOSE YOUR TI-99/4A!!

Here's a short but effective little program for a rapid 'fault' diagnosis on the TI-99/4A. Simply type-in the program in Basic, run it and obey the instructions for a quick analysis of the primary systems of your TI-99/4A. You may get some interesting results!

```

10 RANDOMIZE
20 CALL CLEAR
40 PRINT "THIS IS A"
50 PRINT "COMPUTER DIAGNOSIS!"
60 PRINT
70 PRINT "PRESS (ENTER) AFTER"
80 PRINT "EACH REPLY."
90 PRINT
100 PRINT "WHAT IS YOUR CODE NAME?"
110 INPUT N$
120 IF N$="" THEN 100
130 CALL CLEAR
140 PRINT "ALRIGHT, ";N$;" "
150 PRINT "YOUR CODE NAME IS ACCEPTED."
160 PRINT "NOW, TYPE A FOUR DIGIT CODE"
170 PRINT "NUMBER, THEN PRESS (ENTER). "
180 PRINT
190 INPUT "WHAT IS YOUR NUMBER?" :N
200 IF N>999 THEN 210 ELSE 220
210 IF N<10000 THEN 260
220 PRINT
230 PRINT "INVALID NUMBER! TRY AGAIN."
240 PRINT
250 GOTO 190
260 CALL CLEAR
270 PRINT "YOUR NAME IS: ";N$
280 PRINT
290 PRINT "YOUR NUMBER IS: ";N
300 PRINT
310 PRINT "IS THAT RIGHT? (Y/N)"
320 CALL KEY(0,A,ST)
330 IF ST=0 THEN 320
340 IF A<>89 THEN 20
350 CALL CLEAR
360 PRINT "O.K., ";N$;" , PRESS (C) TO"
370 PRINT "BEGIN A QUICK"
380 PRINT "HARDWARE CHECK."
390 CALL KEY(0,A,ST)
400 IF A<>67 THEN 390
410 CALL CLEAR
420 FOR T=1 TO INT(8*RND)+5
430 PRINT "LOCATION#";T;"= ACCEPTABLE"
440 FOR D=1 TO 300
450 NEXT D
460 NEXT T
470 PRINT "LOCATION#";T;"= ERROR DETECTED!!!"
480 PRINT

```

```

490 PRINT "PRESS (ENTER) FOR HELP"
500 CALL KEY(0,KEY,STATUS)
510 IF KEY<>13 THEN 500
520 CALL CLEAR
530 PRINT N$;" , DIAGNOSTICS SHOW"
540 PRINT "THAT YOUR COMPUTER HAS A"
550 FOR X=1 TO 27
560 READ P
570 PRINT CHR$(P):
580 FOR D=1 TO 200
590 NEXT D
600 NEXT X
610 PRINT
620 PRINT "PLEASE WAIT 15 SECONDS FOR A"
630 PRINT "COMPLETE DIAGNOSIS."
640 FOR D=1 TO 700
650 NEXT D
660 CALL CLEAR
670 FOR T=15 TO 1 STEP -1
680 PRINT "TIME:";T
690 FOR D=1 TO 200
700 NEXT D
710 CALL CLEAR
720 NEXT T
730 CALL CLEAR
740 CALL SCREEN(10)
750 REM **BE SURE YOUR ALPHA LOCK KEY IS DEPRESSED!**
760 A$="FFFFFFFFFFFFFFF"
770 CALL CHAR(128,A$)
780 READ A,B,R
790 IF A=-1 THEN 820
800 CALL HCHAR(A,B,128,R)
810 GOTO 780
820 READ A,B,R
830 IF A=-1 THEN 860
840 CALL VCHAR(A,B,128,R)
850 GOTO 820
860 GOTO 860
870 DATA 83,69,86,69,82,69,32,73,78,84,69,82,78,65,76
880 DATA 32,77,65,76,70,85,78,67,84,73,79,78,2,3,5,2
890 DATA 9,5,2,15,5,2,21,5,6,3,5,6,9,5,6,15,5,8,16,1
900 DATA 9,17,1,10,18,1,11,19,1,11,21,5,11,27,4,14,3
910 DATA 5,14,9,5,14,15,5,14,27,4,18,3,4,18,27,4,23,9
920 DATA 5,23,15,5,23,21,5,23,27,4,-1,-1,-1,2,3,10,2
930 DATA 7,10,2,9,10,2,13,5,2,15,10,2,19,5,2,23,10,2
940 DATA 27,10,14,3,10,14,9,10,14,13,10,14,15,10,14
950 DATA 19,10,14,21,10,14,27,5,18,30,5,-1,-1,-1

```

From THE MSP99 NEWSLETTER Vol 3 No 4
 APRIL 1984
 MSP 99 USERS GROUP, ST PAUL, MINNESOTA

□ END

The first thing I notice with most programmes that use files, is that they usually give the user the choice of cassette or disk. This works reasonable well with files saved on disk as any inefficiencies do not really show up, because disks are fast compared with cassetts. But with most programmes with a disk or cassette option, you can pack and eat a cut lunch whilst the files are being read or saved by a cassette.

To try and speed things up it is important to understand the options you are allowed with cassettes. Firstly the files can be DISPLAY type or INTERNAL. DISPLAY is what it means, the data is saved to the file exactly as it would be printed to the screen. That means that comma's and semi-colon's are print seperators not item seperators. We come now to INTERNAL type records, the preferred type for cassette files as they are much easier to format using print seperators and pending prints. Next the OPEN mode, with cassette you can only specify INPUT or OUTPUT. Lastly the type and length of the files. With FIXED lengths of 64, 128 or 192 bytes. With 64 being the default if you do not specify a length. Now to put this all together.

Firstly we always use INTERNAL type records of 192 bytes long where possible, this means you must be carefull in designing your programme. You may have to limit the length of the item(s) to enable you to fit a certain number of items in a 192 byte record. Now the number of bytes a string uses, is the length of the string plus 1 byte for the length of the string. A number always uses 8 bytes plus 1 byte for its length. So from this we can see it is better to convert numbers to strings with the str\$ function, for unless they are longer than eight digits you save space that way. Then using all those ideas, you then store you records in the PRINT BUFFER using PENDING PRINT statements until the record is 192 bytes long, or as close as possible to that. It does not have to be exact as the computer will pad out the record to the length that was specified. Let us say you keep the records for some sort of sporting club, and you

keep on file the members names and a score. You could nominate to allow 15 letters for the name, and knowing that the scores will not exceed three digits. That gives you 16 bytes needed for the name, and if you convert the score to a string, 4 bytes for the score. All together 20 bytes which means you could load 9 sets of names and scores into the PRINT BUFFER using a pending print type statement. The computer would print 9 sets of names and scores on to the tape at a time. If there were 60 members, the computer would only have to print 7 lots of information on the tape not 60 little items. The programme to do this could look like this.

```
500 N=60
510 OPEN
#1:"CS1",INTERNAL,OUTPUT,FIXED,192
520 FOR I=1 TO N STEP 9
530 FOR J=0 TO 7
540 IF I+J=N THEN 570
550 PRINT #1:
NAME$(I+J);SCORE$(I+J);
560 NEXT J
570 PRINT #1:
NAME$(I+J);SCORE$(I+J)
580 NEXT I .
```

Notice the semi-colon after SCORE\$(I+J) in line 550. That is the PENDING PRINT, when the computer reads that it stores the data in the buffer until the next print statement, and it keeps doing that until we get to line 570 which has no comma or semi-colon at the end of it. So the computer takes that item and all the items it has stored in the buffer and prints them out to the tape in the one go.

The last hint I have, is to have a set number of files so you do not have to save a value for the loop counter. As nothing slows cassette files down more than having a lot of starts and stops, because the start tone uses up as much time as the data loading. Just to give you some idea of how fast cassette files can be, I have a programme that I wrote which saves or reads 300 seperate data items in approximately two and a quarter minuts. So don't write off using cassettes for keeping your files if that is the only system you have, a little bit of thought and planning can work wonders.

See Fred, Wilma and Dino in
this short program that will
run in X-Basic only.

100 ! THE FLINTSTONES

BY STEVEN GERBER

110 !

FROM TICHUG - AUG 85
VIA SYDNEY NEWS DIGEST

120 CALL CLEAR :: CALL MAGNI
FY(3):: CALL SCREEN(15):: DI
SPLAY AT(12,5):"PLEASE WAIT
A SECOND"

130 CALL CHAR(128,"00030F011
E111827424C505F2323130800E0F
F53A0E4F0B03040201828582840B
")

140 CALL COLOR(2,3,3,9,6,6,3
,7,7,12,2,2)

150 CALL CHAR(132,"000000000
00E06183B332F201C1C0A0700000
000101014FCFBDFE7C7A7C78F0"
)

160 CALL CHAR(136,"081010244
484A7AABAA2AABAA2AABAA208080
4242424E4242A2A222A29214949"
)

170 CALL CHAR(140,"0F0F0F1F3
FFFFFFFFFFFFFFFFFFFFFFFF0F0F
8F8F8F8F8F8FEFEFEFEFFFFFFFF")

180 CALL CHAR(120,"000000000
00000010101010101010100000
00000000C0C0C0C0C0C0C080808
0")

190 CALL CHAR(116,"A285AAA6B
1C46A11040404043C20203F4149A
9C10145AB1910101010101010F0"
)

200 CALL CHAR(112,"FFFFFFFFF
FFF3B0000000000000000000FFFFF
FFFFFFFF10000000000000")


210 CALL CHAR(108,"000000000
0000007070707073F3F3F00000
00000000F0F0F0F0F0F0F0F0")

220 CALL CHAR(104,"000100314
A84818141311111090D0300F8048
2017E820A01020C100C04089090"
)

230 CALL CHAR(100,"000000000
0000000000000000000000000000
00000FCFCFEFEFEF0F0F0F8F0F0")

240 CALL CHAR(40,"000000003B
FFFFFF3F0F0F0F0F0F000000F8FC
FFF0000000000000")

250 CALL CHAR(44,"050301070B
1021205F5096110C324380408000
804020207F8105010284788000")



```
260 CALL CHAR(48,"0000000000  
000000000F0F0F0F010000000000  
0000000000FFFFFFFFCF8000000")
```

```
270 CALL CHAR(52,"00000000F  
0F1F1F2020F0F0F0F08000000000  
0080C0C0C00000000000000000")
```

```
280 FOR E=3 TO 10 :: CALL HC  
HAR(E,1,96,32):: NEXT E
```

```
290 FOR I=21 TO 24 :: CALL H  
CHAR(I,1,125,32):: NEXT I ::
```

```
FOR C=10 TO 19 :: CALL HCHA  
R(C,1,40,32):: NEXT C
```

```
300 FOR D=9 TO 15 :: CALL HC  
HAR(D,10,32,10):: NEXT D ::
```

```
CALL HCHAR(8,8,32,14):: FOR  
F=10 TO 15 :: CALL HCHAR(F,1  
7,48,2):: NEXT F
```

```
310 FOR G=10 TO 11 :: CALL H  
CHAR(G,12,96,2):: NEXT G ::
```

```
FOR H=6 TO 7 :: CALL HCHAR(H  
,18,32,1):: NEXT H
```

```
320 DISPLAY AT(1,8):"THE FLI  
NSTONES" :: DISPLAY AT(2,7):
```

```
"BY STEVEN GERBER"
```

```
330 CALL SPRITE(#4,121,12,12  
6,110,#5,130,2,110,110,#6,13  
4,12,110,110,#7,138,2,126,11  
0,#8,142,10,126,110)
```

```
340 CALL SPRITE(#9,117,2,142  
,110,#10,113,10,142,110,#11,  
108,12,142,110,#12,105,2,73,  
90,#13,101,12,73,90)
```

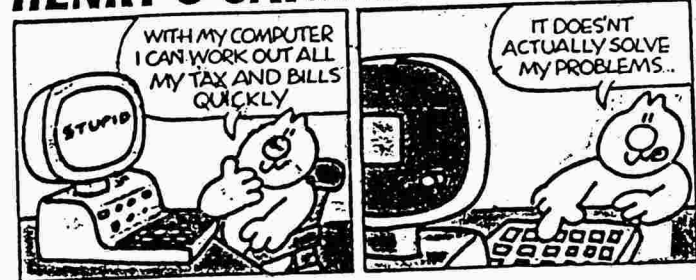
```
350 CALL SPRITE(#14,41,10,73  
,90,#15,45,2,85,153,#16,49,1  
1,85,153,#17,53,6,85,153)
```

```
360 GOTO 360
```

FROM TIT-BITS VOL 4 NO 3 FEB 1986

--TI Users PERTH (Australia)--

HENRY'S CAT STAN HAYWARD & BOB GODFREY



--- from
'Sunday'
magazine

* EARLY BASIC SERIES *

By Garry Jones

The following explanations are for words you may find in the USERS REFERENCE GUIDE or THE BEGINNERS BASIC Book and may have trouble comprehending.

PERIPHERALS- Anything which is attached to the Computer.
eg. Printer, Speech synthesizer, Expansion Box etc.

ASCII CODE- ASCII means American Standard Code for Information Interchange. The Code is a different numeric value for all Alphas, numbers and symbols.

BINARY- The Computer's Alphabet. Because the Computer is only a large box of switches and all depending on how these switches are set in Memory is how the Data is stored, calculated and processed. Thus as a switch is only an ON or OFF device and the Computer is similar in operation, the Computer's alphabet needs only two characters. These are "1" and "0" and the alphabet is known as BINARY.

BYTE- Usually 8 bits or Binary numbers to make up a byte. The Computer is often rated in the amount of storage of the bytes in R.A.M. eg. 16,384 bytes or 16K for the T.I.99/4A.

CURSOR- The flashing square which indicates where the next character will appear on the screen, also indicates the Computer is ready to use or accept data.

DATA- The information handled or produced by the Computer.

HEXADECIMAL- Is a 16 number base system (0-9 and A-F) which is a shorthand way of writing binary.
eg. 0110=6, 1011=7, 1111 0110=F6

INTEGER- A whole number. If used as a function this will round off to the lowest whole number.
eg. 1.5=1, 1.99=1, -3.1=-4, -4.9=5

PROGRAM- A set of instructions which tell the Computer how to perform a Task.

STRING- A series of Alphas, numbers and symbols treated as one unit inside quotation marks.

NULL STRING- A string which contains no characters and has a zero length.

COMMAND- An instruction performed immediately with no line number needed.
eg. RUN, NEW, OLD, SAVE, PRINT

STATEMENT- AN instruction that is preceded by a line number.
eg. GOTO, GOSUB, IF/THEN

FUNCTION- Allows a number of steps to be achieved under a single instruction.
eg. ASC< INT< SGN< SIN< SQRT

PROGRAM LINE- A line that contains a statement or statements for the Computer's operation.

HARDWARE- The following hardware devices make up a Computer System: MONITOR, DISK DRIVE, PRINTER, MEMORY EXPANSION.

SOFTWARE- Any type of program used by the Computer which includes programmes in R.O.M.

LOOP- Repeats a part of a program a number of times, usually a specified number.

INPUT- Is data that is being accepted from an external device or the act of accepting data from an external device.
eg. KEYBOARD, MODEM, CASSETTE.

OUTPUT- Is data being supplied by the Computer or the action of supplying data from memory onto a screen, printer, cassette, or disk.

BUG- An error or defect in a program which causes a mis-operation or halt to the program.
eg. Debugging is removing such faults.

SUBPROGRAM- A procedure usually stored in the R.O.M. of the Computer and is accessible via a CALL instruction.
eg. CALL CLEAR, CALL HCHAR, CALL VCHAR

SUB ROUTINE- Is part of a program that has been loaded into the Computer which can be used many times. Is accessible via GOTO, GOSUB, and RETURN.
eg. complex calculations.



EDITED FROM

The Ottawa T.I. 99/4A Users' Group NEWSLETTER

A LOOK AT MEMORY by R. A. Green

In this article we will have a look at the memory system of the TI 99/4A. You have all heard the terms: ROM, GROM, GRAM, RAM used constantly in conversations at the Users' Group meetings. Maybe you are confused by all this jargon, or are not quite sure what they all mean. I hope things will be clearer by the end of this article.

To begin, there are three different kinds of memory in the TI. They are: (1) CPU memory, (2) Video Display memory, and (3) Graphics memory.

CPU memory is the most important -- it's the only memory that belongs to the 9900 microprocessor (the CPU). All machine language programs must be in CPU memory before they can be executed by the CPU.

Video Display memory is the memory used mainly to generate the TV picture. It belongs to the Video Display Processor (VDP). Video Display memory cannot be accessed directly by the CPU. The CPU must ask the VDP to fetch or store bytes in the Video Display memory.

Graphics memory is the memory where the Graphics Programming Language (GPL) programs are stored. Like the Video Display memory, Graphics memory cannot be accessed directly by the CPU. The CPU must ask for bytes to be fetched or stored one at a time. This type of memory is unique to TI. I assume it was intended to help prevent -- dare I say it -- pirating.

Now we have looked at the three kinds of memory in the TI 99/4A. Each of the three kinds of memory could exist in two forms: (1) READ ONLY, which cannot be destroyed, and (2) READ WRITE, which can be changed.

Now, we have a small problem with terminology. Originally, in the computing field, RAM meant Random Access Memory, however, since the advent of micro/home computers the term has come to mean Read/Write Random Access Memory. The term ROM is used for Read Only Random Access Memory.

We have taken a few steps along the way to understanding and now is the time for a little table to organize our thoughts.

#	OWNER	TYPE	COMMON NAME
1	CPU	ROM	ROM
2	CPU	RAM	CPU RAM
3	VDP	RAM	VDP RAM
4	GRAPHICS	ROM	GROM
5	GRAPHICS	RAM	GRAM

To refine our terminology a bit, let's look at the common terms used by the TI enthusiast.

- ROM -- means the CPU ROM
- RAM -- used, mainly by TI in its advertisements
means VDP RAM, which is the term I try to use
- CPU RAM -- means the CPU RAM (strangely enough)
- GROM -- means Graphics read only memory
- GRAM -- means Graphics read/write memory
like in the GRAM KRACKER or MAXIMEM

A LOOK AT MEMORY

Remember that the CPU, the 9900 microprocessor can only execute programs in CPU ROM or CPU RAM. Anything in the other kinds of memory is just data for some program executing in the CPU.

Now we know the five kinds of memory in the TI 99/4A, and we have the terminology to talk about them. This is a good time to ask, "Where is all this memory?"

CONSOLE (bare bones with no expansion)

CPU ROM - 8K bytes
CPU RAM - 256 bytes
VDP RAM - 16K bytes
GROM - 18K bytes

CARTRIDGE (if it were full)

CPU ROM - 8K bytes
GROM - 30K bytes

MEMORY EXPANSION

CPU RAM - 32K bytes

I/O DEVICES (RS232 and Disk)

CPU ROM - 8K bytes

Now that we are talking sizes, let's talk about the maximum sizes. As we all know (and like to brag about), the TI 99/4A is a 16 bit machine. This means that it can count (or address) bytes from 0 to 65536 (64K). So we say it can address 64K of memory (i.e. CPU memory). The TI 99/4A also uses a 16 bit word to tell the VDP which byte it wants. Two bits of this word are used as indicators so that there is only 14 bits of address. So the maximum VDP memory is 16K bytes. A 16 bit address is also used for Graphics memory (with no indicators) so its maximum is again 64K.

We are now ready to take another step, and find out what is in all this memory.

CPU ROM contains 3 main things, all written in assembler:

- The low level operating system. This code is necessary to make the machine do anything.
- The interpreter for the GPL programs.
- The Device Service Routines (DSR) which provide high level access to the input/output devices.

GROM contains 3 main things:

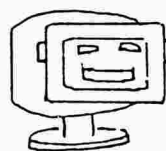
- The high level operating system, written in the GPL language. The high level operating system gives you the master title screen, the first level menus and provides for calling other programs written in the various languages.
- The TI BASIC interpreter, written in GPL.
- Tables of data for initializing the VDP.

VDP RAM contains 3 main things:

- The TV screen image, character, colour and sprite tables.
- Work areas for programs and device service routines.
- BASIC programs, which are interpreted by the TI BASIC or EXTENDED BASIC interpreter.

CPU RAM contains 3 main things:

- Work space for the GPL interpreter (the 256 bytes that are in the console).
- Assembler language programs.
- EXTENDED BASIC programs, which are interpreted by the EXTENDED BASIC interpreter.



BASICally ---

--- a piece of cake!

Does the term 'variable' send shudders up your spine? It did mine, until I got down to the nitty-gritty of learning Basic on the old TI. For a week my table was covered with Tex and text - the only things to eat were paper, or QWERTY keys! I was forced to "digest" those variables

THE COMPUTER:
ALWAYS AT
YOUR SERVICE



So, think of the processing part of the computer as a collection of "memory boxes", like the proverbial pigeon holes, each one of these having its own unique label, or address, (which is the correct term). The first time we want to reference a piece of information in our program, be it a number or a 'string' of characters, we will give it a name. This variable name becomes the label on one of these pigeon holes, and when the program is run, the value that we assign to that variable name will be put inside its own pigeon hole. This value can be called up many times during a program by using its variable name.

Variable names can start with a letter, @, /, [,], or _ . The only characters allowed in them are letters, numbers, @ and _ . The one exception is the \$ sign: all string variable names must end in \$. Variable names can be up to fifteen characters long, including \$. It's a good idea to use variable names that remind you of what you're representing in the program:

Examples of some Numeric Variable Names:

CAR1, AB23, MAGGIE_THATCHER, FIRST, A

Examples of String Variable Names:

CAR1\$, AB23\$, GAMES\$, L\$, N1\$, XY\$

Reserved words, such as RUN or LIST, cannot be used by themselves, but can form part of a variable name, such as LIST\$, RUNNER.

Numeric Variables

Using variable names we can call up the values in our pigeon holes and make use of the computers number-crunching capabilities. Let's get it to add two numbers together and print the result:

BASIC BEGINNINGS - VARIABLES

(we'll give our numbers the names A and B):

```
10 LET A=10          (sets up a box called 'A' and puts a value of
20 LET B=20          ten in the box; does the same for 'B', with
30 PRINT A+B         its value of twenty.
                    Looks up box 'A', gets out its value and puts
                    it in an adding register; does the same for
                    'B'; adds them together and displays the
                    result on the screen)
```

The values of A and B remain in the boxes unchanged, unless we choose to overwrite them. We could say:

```
LET A=11 or LET A=A+1 The value in the box 'A' is now 11.
```

The computer uses '+' and '-' as we do in arithmetic, but for 'x' it uses '*' and for '÷' it uses '/' .

String Variables

Groups of characters "like these" are called "strings" because they can be strung out like beads on a string, and can be up to 255 characters long! Their variable names must end in \$ and the string itself is put into inverted commas, so the computer will treat it as a unit:

```
10 LET Q$="HELLO"    (Q$ is the variable name, with value
                    "HELLO")
```

Strings allow us to process textual information and can be any mixture of characters on the keyboard, even a space is a character. They can be cut up, joined together and counted using the string operators. As with numbers, think of the value of the string variable as being put into the box with its name on; it can be called up in the program by using its name. Here are two of the operators, and their functions, I'll detail the rest next time.

1. & To join strings together use ampersand, "&"

```
For example:      10 A$="CON" & "CAT" & "EN" & "ATION"
                  20 PRINT A$
```

This will print out CONCATENATION which means 'make a chain', join together.

2. SEG\$ Cutting up strings is simple with TI BASIC. SEG\$ does this, but you have to tell it: a) which string to look at; b) which character to start counting from; c) how many characters to count.

```
Try this: 10 A$="SHOE STRING"
           20 N$= SEG$(A$,6,6)      (count in A$, from the sixth
           30 PRINT N$             character, six characters
                                   along)
```

Prints out STRING on the screen.

GETTING IT RIGHT WITH YOUR TI-WRITER

by Bill Browning

PART ONE

This is the first in a series of articles on ideas for increasing the efficiency of your TI-Writer program. How many articles in the series will depend on how many ideas I have.

In my work as a writer, I have made extensive use of the TI-Writer program. I'd like to share with you some methods I have developed for saving time and energy. Also, I want to invite you to tell me about your ideas, to be included in future articles. But more about that later.

If TI-Writer is the only word-processing program you have ever used, you may be wondering how good it is. Let me begin by saying that, among programs designed for 48K of memory, it is top notch. It was described by one reviewer (InfoWorld, Vol. 5, Nr. 42) as a "heavy hitter" among word-processing programs.

For example, its speed is extraordinary. Where most programs paint the words on the screen, TI-Writer splashes them on with no delay. And the amount of available memory for text is larger than I've seen for other 48K programs with similar features. The manual that you received with the software also is excellent. However, to paraphrase Johnny Carson, it does not include everything you'd ever want to know about using the system. In writing these articles, I'm assuming that you have read the manual. I'm going to try to avoid repeating information included there.

So let's get down to cases:

CHALLENGE: If someone were to publish a "Ten Most Unwanted Features" list for the 99/4A, high on the list would be the inconvenient location of the apostrophe, question mark, and quotation mark keys. For these frequently used keys, you press the function key instead of the more logical shift key.

For certain writing jobs, such as dialogue, where you may be using several quotation marks per line, this inconvenience can become very frustrating and time consuming.

SOLUTION: Global Search to the rescue! When you know you will be using one of these inconvenient keys a lot, select a more convenient, substitute key that you know you won't be using during that writing.

I've found that the equals (=) sign works well for this substitution because you seldom use it in non-math writing. Also, it is much more convenient than the " key. The = key can be pressed without having to press any other key.

The procedure is very simple: As you're composing, key in = wherever a " belongs. For example: He turned to her and said, =I don't agree.=

Then, after completing your text and before printing it out, press FUNCTION + 9 (Command mode); S for Search; ENTER; 1 (for line 1); then press ENTER again. This sequence puts the cursor at the top of the file, so that the entire file will be searched for replacing each = with a ". Press FUNCTION + 9 again; Key in RS for Replace String; and press ENTER.

Next, key in: /=/" and press ENTER.

When the cursor reaches the first instance of a = sign, press A for All, sit back, and let the computer do the work. Within a few seconds, all instances of = will have been changed to ".

Save the file to disk, and print it out.

You might think that the slash (/) key would be even more convenient than the = key. Unfortunately, since slashes are used as dividers during the Replace-String process, the computer does not recognize them for replacement. In short, slash the / as a potential substitute key.

One possible goof would be forgetting to replace the = signs before printing out. To avoid this error, I suggest that you place a (very large) sign on your printer saying, =Don't forget the quotation marks, stupid! = (Oops. I forgot the quotation marks.)

Naturally, you can use this method to substitute for other keys as well. Just make sure the substitute key isn't going to be used for itself anywhere in the document.

From THE MSP99 NEWSLETTER Vol 3 No 4
APRIL 1984
MSP 99 USERS GROUP, ST PAUL, MINNESOTA



by Bill Browning

PART TWO

This is the second in a series of articles on ways to increase the ease and efficiency of your TI-Writer word-processing program.

In the first article, I talked about a way to use Replace String to make your work easier. This article covers another use of that handy feature.

Word-processing programs can be grouped into two main categories: (1) symbol formatted, and (2) what-you-see-is-what-you-get. With the symbol-formatted kind, the user embeds symbols in the program to tell the printer how to format the text. With the what-you-see kind, the screen displays the text in essentially the same way it will appear in print.

TI-Writer is somewhat unusual because you can use both methods. "Fill" (.FI) is equivalent to symbol formatting, and "No Fill" (.NF) is a "what-you-see" method.

The what-you-see method usually has an 80-column format, in order to accommodate the line width found on a typical 8-1/2 x 11 sheet. But if you want 80 columns on TI-Writer, you have to look at them section by section in three overlapping windows. Most users find this process to be extremely bothersome, because when reviewing what you have written, you cannot see an entire block of material at one time.

To avoid windowing, you can use the .FI, or symbol formatting, mode. You simply set up the left and right margins on the TAB line to fill one screen's width. For example, for a typical TV screen, you might set the left margin at "2" and the right margin at "39." This setup gives you a 38-column display. On some sets, a slightly narrower or wider display may be optimal. (For simplicity, I will refer to this as a 40-column format.)

You then include formatting symbols that tell the printer to print whatever margins and indentions you desire. (Of course, your manual explains how to do that.)

However, the 40-column approach has a limitation that can be serious in some cases: By looking at the screen, you cannot tell where page breaks (page endings) will occur. Therefore, without printing the document to see where the page breaks are, you have no way of eliminating widow lines.

A widow line isn't something you find at the local welfare office. It's an isolated word or line at the top or bottom of a page. It may be a subheading separated from the following material, or it may be a line isolated from the rest of a paragraph. In any case, because widow lines look awkward, they should be avoided if possible. Therefore...

THE CHALLENGE: How to use the 40-column format to avoid windowing, and yet be able to preview the material on the screen to insert page breaks that will eliminate widow lines in your printout.

We are assuming that (1) the document is too long to be quickly printed out in draft form, and (2) the document is important enough for you to be concerned about page breaks.

THE SOLUTION:

1. Key in your text using the 40-column format for ease of typing and editing. But do not include any margin (such as .LM 12) or centering (.CE) format commands, because you will be converting the format to a No Fill (NF), what-you-see display.

At the beginning of each paragraph, key in a symbol that will not be used elsewhere in the text. A good symbol to use is the >. The reason for doing this will be made clear in a moment.

Do not indent the paragraphs at this stage. Below is an example of what the beginning of a paragraph would look like:

>Now is the time....

2. When you are ready to print out the copy, first convert to a what-you-see format. To begin with, change the left and right margins on the Tab line to the actual ones you want on the printout.

Then, use the Replace String mode to automatically reformat the document: If the paragraphs are not to be indented, use />/. If they are to be indented, use />/^ ^ ^/, with one space symbol (^) for each indentation space.

Press "ENTER" and "A" for All, when that prompt appears. Then, sit back. The file will be automatically converted from 40 columns to the actual printing width.

3. Save the file under a different name, so that you don't lose the original 40-column format, in case you want to use it later for more typing or editing. (One handy notation is to put NF, for No Fill, on the end of the new file name, to distinguish it from the other one.)

CONTINUED →

4. Starting at the beginning of the new file, go through and type in forced page breaks (.BP) where you want pages to end. The following table will help you determine page breaks. The table assumes standard 11 inch long paper:

SPACING	MAXIMUM LINES OF TEXT
Single	58
1-1/2	36
Double (.LS 2) . .	29

For example, if you are using single spacing, you will want to make page breaks at or before 58 lines per page (including skipped lines). These numbers hold true whether or not you are using a header and/or footer.

When using 1-1/2 spacing (assuming your printer can support that spacing) set the page length (.PL) at 44 instead of the default of 66.

When you are using .LS 2 for double spacing, the printer skips 2 spaces instead of 3 when you skip a line while typing. Use the following formula when your double spaced text includes skipped lines:

$$29 + (\text{Skipped Lines}/2) = \text{Line Count}$$

In other words, count all the skipped lines in a 29-line block. Divide that number by 2. Add that number to 29. Make the page break no later than the line following the sum.

For example, if there are 6 skipped lines in a 29-line block, divide 6 by 2 and add the result, 3, to 29. Thus, you would put a forced break following line 32, or before that if necessary to avoid a widow line. (You really didn't want to use double spacing after all, did you.)

The line numbers in the TI-Writer program are very helpful for locating page breaks. So is a calculator.

5. Save the file with the forced page breaks, and print it out.

On a long document, this process will take a bit of time. (Less time, however, than required to explain it.) But it does give you a way to preview the document as it will appear on the printed page.



From THE MSP99 NEWSLETTER Vol 3 No 5
MAY 1984
MSP 99 USERS GROUP, ST PAUL, MINNESOTA

ATTENTION ASSEMBLY LANGUAGE PROGRAMMERS HAVING PROBLEMS ?

EXTRACTED FROM TIT-BITS Vol 3 No 4
TI-USERS PERTH (Australia)

They could be caused by the E.I.S. (Enhanced Instruction Set) recently implemented on the TI 99/4A.....

Listed below are the mnemonics followed by a brief description.

BH	Branch and Hang
HCF	Halt and Catch Fire
BOI	Byte Operator Immediately
CDS	Condense and Destroy System
CLBR	Clobber Register
BOB	Branch On Bug
CRN	Convert to Roman Numerals
CPPR	Crumple Printer Paper and Rip
EIOC	Execute Invalid Op Code
EPI	Execute Programmer Immediately
EDU	Execute Dissatisfied User
IBP	Insert Bug and Proceed
RPM	Read Programmer's Mind
RSD	On Read Error Self Destruct
MET	Misread and Eat Tape
STT	Stretch and Tangle Tape
SRSD	Seek Record and Scar Disk
ED	Eject Disk
BTAD	Branch To Auto Destruct
JKL	Jump to Random Location
JSP	Jump on Sexy Programmer
BSO	Branch on Sleepy Operator
RCI	Read Card and Ignore
RASC	Read And Shred Card
RCS	Read Card Sideways
DO	Divide and Overflow
DC	Divide and Conquer
FCE	Fill Core with Epoxy
CCP	Clear Core and Proceed
PCB	Pause for Coffee Break
SPD	Start and Power Down
UER	Update and Erase Record
WRTC	Wind Real Time Clock
WWTC	Wind Wrong Time Clock
RWOM	Read Write-Only Memory
WROM	Write Read-Only Memory
BTI	Blame TI. (Annon...)

With Acknowledgments to 'TACONEWS'
May/June 1983.

P. West T.I.U.P.

Here's a little TI WRITER tip, on how RAMBLES is put together using the formatter.

We start with a disk of assorted files in normal DV80 format which need formatting. Instead of adding formatter instructions to each file, I add a formatting file to the disk, which looks like this:

```
.LM 0;RM 54;FI;AD;PL 600
.IF DSK2.*1*
.PL 1
```

and save this as DSK1.FORMAT

Then I switch to FORMATTER

The input file is of course DSK1.FORMAT

The output I direct to disk, for cleaning up, for instance, output to DSK1.R1

Then keep pressing ENTER

Then the *1* in our FORMAT text produces a prompt at the bottom of the screen, and we type in the name of the file on DKS2 than we want to format.

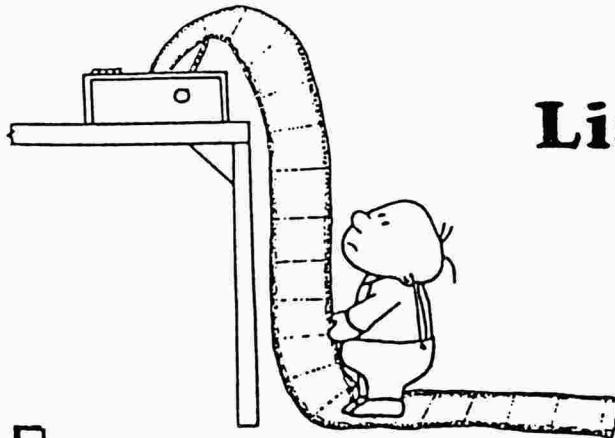
OFF it goes.

At the end we just reselect FORMATTER, change the name of the output file, and put in the name of the next file to be formatted at the prompt

Stephen Shaw

Program

Listing



HEADING FROM TIT-BITS
TI-USERS PERTH (Australia)

TI-TIMER

```

100 CALL CLEAR
110 REM *****
    *   TI-TIMER   *
    * BY IAN HAKANSON *
    * EXTENDED BASIC *
    *****
120 REM
    THE PROGRAM DEMONSTRATES
    THE USE OF A SPRITE TO
    TIME PROGRAM EXECUTION
    AND PROMPT RESPONSES.
130 REM THE TIMER IS SLOWED
    DOWN IF THERE ARE MANY
    CALCULATIONS, SQUARE
    ROOTS OR TRIG FUNCTIONS
    IN THE PROGRAM.
140 REM THE TIMER IS ALWAYS
    ACCURATE FOR PROMPT
    RESPONSES.
150 REM TIMES UP TO A
    MAXIMUM OF 10 MINUTES
    56 SECONDS CAN BE READ.
.....
160 REM =====
    EXAMPLE SHOWING
    CALCULATION OF PROGRAM
    EXECUTION TIME.
=====
170 CALL START_TIMER
180 FOR I=1 TO 3000 :: K=1 :
: NEXT I
190 CALL READ_TIMER(TIME)
200 DISPLAY AT(5,1):"EXECUTI
ON TIME =";TIME;"SECONDS"
210 REM =====
    EXAMPLE SHOWING
    CALCULATION OF PROMPT
    RESPONSE TIME.
=====

```

Here is a short program from Ian Hakanson that uses sprites to act as an accurate timing device. It is an accurate method of determining the elapsed time of an event. The REM statements put in by Ian, explain the operation of the program and the timing. Thanks Ian for this little snippet of useful information.

```

220 DISPLAY AT(12,1):"ENTER
VALUE"
230 CALL START_TIMER
240 ACCEPT AT(12,13):V
250 CALL READ_TIMER(TIME)
260 DISPLAY AT(14,1):"RESPON
SE TIME =";TIME;"SECONDS"
270 GOTO 220
280 END
290 REM #####
    ROUTINE TO START TIMER.
#####
300 SUB START_TIMER
310 CALL SPRITE(#1,143,1,256
,256,1,1):: SUBEND
320 REM #####
    ROUTINE TO READ TIMER
    TO NEAREST SECOND.
#####
330 SUB READ_TIMER(T)
340 CALL POSITION(#1,Y,X)::
D=X-(Y+(Y>194)*32):: IF D>0
THEN P=(224-D)/32 ELSE P=ABS
(D/32)
350 T=INT((P*256+X)/3.12+.5)
360 SUBEND

```

After the excitement of the Derby meeting and the election of the new committee and because the day passed all too quickly, I was left slightly "jet lagged" and wondering how best to tackle my new position so I shall outline some ideas and hopefully receive some feedback from you the reader as to what projects or services will be best suited.

I will where possible evaluate all third party projects and hopefully provide a backup kit facility (more of which later), that is if a project appears in the magazine all parts will be available either from myself or a third party.

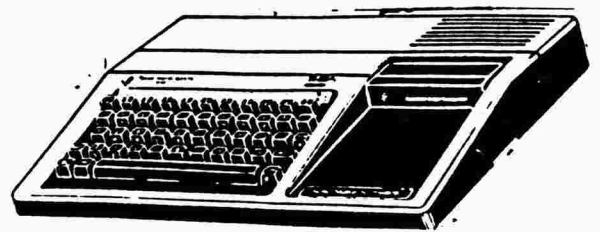
I would like to form a register of projects so that we can keep up with developments and hopefully either put people in touch with like interests or stop duplication.

All projects will be compiled into a separate booklet at suitable intervals and larger projects made available as a printout to save bogging the magazine down.

Projects that I have in hand are an organ type keyboard expansion and an EPROM eraser kit. Also several small articles passed on to me by Stephen Shaw one of which will appear in this issue space permitting. It will probably be some time before the larger projects are in print so please get in touch directly if you are interested. Check with Jim Ballinger for my phone number and address as they will be changing shortly.

Any suggestions for projects will be more than welcome but if there is no feedback from you the reader then I will assume that you are delighted with what I produce and it will serve you right! .

Mike



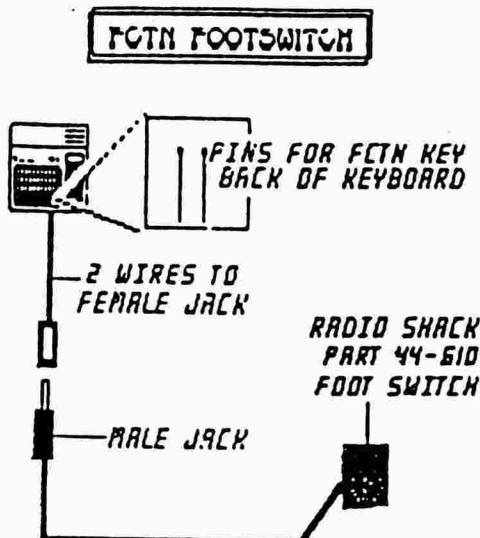
BRUCE'S COMPUTER MAGIC

by BRUCE KALVER



Here is a quick do-it-yourself project for all you TI-WRITER users out there. This is how t attach a footswitch to your computer to use instead of your FCTN key. It is very handy while using TI-WRITER so you don't have to do all those awkward positions to make things work. I learned about this while at a TI computer fair. I was standing around a table waiting for a demonstration on how to clean your computer when a member of the club asked me if I had invented any new gadgets for the TI. The guy behind me overheard this and one thing led to another and this guy told me about his footswitch. I jotted down some notes on a candy wrapper and thanked him. Alas I have misplaced my Cadbury Crunchie Bar wrapper so to the mystery man from Massachusetts, I thank you for the idea. The picture is pretty straight forward. The two wires are soldered on the two pins below the FCTN key then out as described in the picture.

Thanks to all the newsletters that have been reprinting my articles. Please remember to give credit to the North Eastern 99ers who publish my articles first.



25.6.87

FUNCTION FOOTSWITCH NOTES

This article is one passed on by Stephen Shaw. I have not had the opportunity to test it but the idea is so simple that there is no reason it should not work. I cannot find the switch listed in the TANDY UK catalogue but a suitable alternative would be the MAPLIN part no LB 64U footswitch at £4.95.

Mike

SPIDER BOP

From Chicago Times Vol 4 No 5
30th December 1984
Chicago TI-99/4A Users Group

```
100 REM SPIDER BOP 12/19/84
110 REM BY JOHN BEHNKE
120 REM BASIC OR X-BASIC
130 REM DISK DRIVE REQUIRED
140 REM FOR HIGH SCORES
150 CALL CLEAR
160 RANDOMIZE
170 A=0
180 B=12
190 C=0
200 CALL CHAR(96,"1818181818
181818")
210 CALL CHAR(104,"6699997E5
595A5A5")
220 CALL CHAR(112,"3C7EFFFF
F7E3C3C")
230 CALL CHAR(113,"3C3C7EFFF
F7E3C3C")
240 CALL CHAR(97,"000000FFFF
")
250 FOR E=1 TO 8
260 CALL COLOR(E,16,1)
270 NEXT E
280 CALL SCREEN(2)
290 CALL COLOR(9,16,1)
300 CALL COLOR(10,14,1)
310 CALL COLOR(11,3,1)
320 FOR E=1 TO 10
330 D(E)=3
340 NEXT E
350 A$="SPIDER BOP"
360 F=4
370 G=11
380 GOSUB 950
390 F=22
400 G=5
410 A$="USE <ARROW KEYS> TO
MOVE"
420 GOSUB 950
430 F=8
440 G=3
450 A$="WHAT ARE YOUR INITIA
LS? ___"
460 GOSUB 950
470 NAME$=""
480 FOR E=27 TO 29
490 CALL KEY(0,KEY,ST)
500 IF ST=0 THEN 490
510 CALL HCHAR(B,E,KEY)
520 CALL SOUND(10,500,0)
530 NAME$=NAME$%CHR$(KEY)
540 NEXT E
550 F=10
560 A$="SET UP NEW HIGH SCOR
E FILE?N"
570 GOSUB 950
580 CALL KEY(0,KEY,ST)
590 IF ST=0 THEN 580
600 CALL SOUND(10,500,0)
610 IF KEY=13 THEN 630
620 CALL HCHAR(10,30,KEY)
630 CALL GCHAR(10,30,TE)
640 IF TE<>89 THEN 660
650 GOSUB 2120
660 F=24
670 G=3
680 A$="PRESS <ENTER> TO BOP
SPIDERS"
690 GOSUB 950
700 A$="PUNCH POWER:"
710 F=12
720 G=10
730 GOSUB 950
740 A$="(H)I (M)ED (L)OW"
750 F=14
760 G=8
770 GOSUB 950
780 CALL KEY(0,H,I)
790 IF I=0 THEN 780
800 IF (H<>72)*(H<>77)*(H<>7
6) THEN 780
810 IF H<>72 THEN 840
820 J=1
830 GOTO 890
840 IF H<>77 THEN 870
850 J=2
860 GOTO 890
870 IF H<>76 THEN 890
880 J=3
890 GOSUB 990
900 F=1
910 G=17
920 GOSUB 1100
930 GOSUB 1530
940 GOTO 920
950 FOR E=1 TO LEN(A$)
960 CALL HCHAR(F,G+E-1,ASC(S
EG$(A$,E,1)))
970 NEXT E
980 RETURN
990 F=1
1000 CALL CLEAR
1010 G=10
1020 A$="SCORE: 0"
1030 GOSUB 950
1040 FOR E=1 TO 10
1050 CALL HCHAR(3,E*3,96)
1060 CALL HCHAR(4,E*3,104)
1070 NEXT E
1080 CALL HCHAR(24,B,112)
1090 RETURN
1100 A=INT(10*RND)+1
1110 IF D(A)+J>23 THEN 1170
1120 D(A)=D(A)+J
1130 CALL SOUND(-50,110+50*R
ND,0)
1140 CALL VCHAR(D(A)-J,A*3,9
6,J)
1150 CALL HCHAR(D(A),A*3,104
)
1160 RETURN
1170 CALL HCHAR(D(A),A*3,32)
1180 B=B+1
```

```

1190 FOR E=A*3 TO B STEP SGN
(B-A*3)
1200 CALL HCHAR(24,E,104)
1210 CALL HCHAR(24,E,32)
1220 NEXT E
1230 CALL HCHAR(24,E,104)
1240 FOR E=1 TO 400
1250 NEXT E
1260 CALL CLEAR
1270 F=1
1280 G=12
1290 A$="GAME OVER"
1300 GOSUB 950
1310 GOSUB 1870
1320 OPEN #1:"DSK1.SPIDERFIL
E",INTERNAL,VARIABLE
1330 FOR I=5 TO 1 STEP -1
1340 INPUT #1:A1,A1$
1350 F=5+I*2
1360 G=13
1370 A$=A1$
1380 GOSUB 950
1390 G=17
1400 A$=STR$(A1)
1410 GOSUB 950
1420 NEXT I
1430 CLOSE #1
1440 F=20
1450 G=3
1460 A$="PRESS <ENTER> TO PL
AY AGAIN."
1470 GOSUB 950
1480 CALL KEY(0,H,I)
1490 IF I=0 THEN 1480
1500 IF H=13 THEN 150
1510 CALL CLEAR
1520 END
1530 CALL KEY(0,H,I)
1540 IF I=0 THEN 1700
1550 CALL SOUND(-10,-3,5)
1560 IF H<>68 THEN 1620
1570 CALL HCHAR(24,B,32)
1580 IF B<30 THEN 1600
1590 B=0
1600 B=B+3
1610 CALL HCHAR(24,B,112)
1620 IF H<>83 THEN 1680
1630 CALL HCHAR(24,B,32)
1640 IF B>3 THEN 1660
1650 B=33
1660 B=B-3
1670 CALL HCHAR(24,B,112)
1680 IF H<>13 THEN 1700
1690 GOSUB 1710
1700 RETURN
1710 FOR K=24 TO 5+J*2 STEP
-1
1720 IF K=D(B/3) THEN 1760
1730 CALL HCHAR(K,B,112)
1740 NEXT K
1750 GOTO 1830
1760 CALL SOUND(100,-5,0)
1770 CALL VCHAR(K-4+J,B,32,4
)
1780 CALL HCHAR(K-5+J,B,104)

```

```

1790 D(B/3)=K-5+J
1800 C=C+1
1810 A$=STR$(C)
1820 GOSUB 950
1830 FOR E=K TO 23
1840 CALL HCHAR(E,B,32)
1850 NEXT E
1860 RETURN
1870 REM HIGH SCORE
1880 OPEN #1:"DSK1.SPIDERFIL
E",INTERNAL,VARIABLE
1890 FOR I=1 TO 5
1900 INPUT #1:HS(I),NAM$(I)
1910 NEXT I
1920 IF HS(1)>C THEN 2100
1930 HS(1)=C
1940 NAM$(1)=NAME$
1950 FOR I=1 TO 4
1960 FOR J=I+1 TO 5
1970 IF HS(I)<=HS(J) THEN 204
0
1980 CHANGE=HS(I)
1990 NN$=NAM$(I)
2000 HS(I)=HS(J)
2010 NAM$(I)=NAM$(J)
2020 HS(J)=CHANGE
2030 NAM$(J)=NN$
2040 NEXT J
2050 NEXT I
2060 RESTORE #1
2070 FOR I=1 TO 5
2080 PRINT #1:HS(I),NAM$(I)
2090 NEXT I
2100 CLOSE #1
2110 RETURN
2120 REM SET UP NEW FILE
2130 OPEN #1:"DSK1.SPIDERFIL
E",INTERNAL,VARIABLE
2140 FOR I=1 TO 5
2150 PRINT #1:0,"___"
2160 NEXT I
2170 CLOSE #1
2180 RETURN

```

NO DISK DRIVE?
CHANGE THE FOLLOWING LINES:

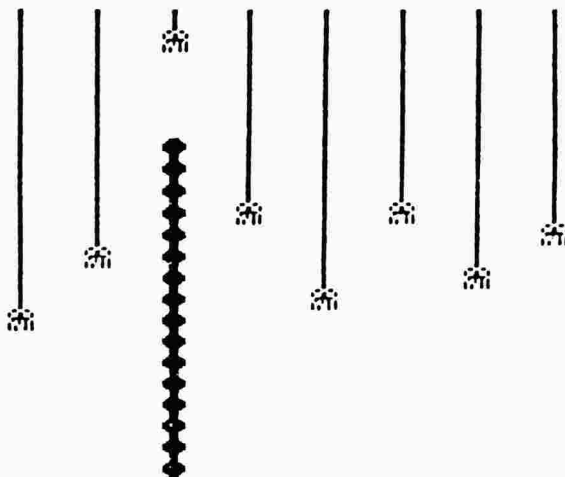
*

```

1320 GOTO 1440
1880 GOTO 2110
2130 GOTO 2180

```

SCORE: 1



TI WRITER

TI WRITER NOTES

EXTRACTED FROM TITBITS
 TI-USERS PERTH (Australia) (TitBITS Vol 2 No 2)
 Author: Bernie Elsner (Oct 83 - June 84)
 (Translation: goog = egg.
 Full as a goog = drunk)

Now, how about a quick quiz question? What is the maximum number of characters you can store in TI-WRITER's EDITOR? If you haven't tried it out have a guess.... 16K?, 32K?, 48K? 22K? 23K? WRONG! Give up? How about 375.5K? IMPOSSIBLE?

Well try it out. Fire up the editor and type up a few 80 column lines of the letter "A" (I didn't say that the letters had to be different did I?) Then use the copy command to keep doubling the number of lines you've got until you get MEMORY FULL. Press enter to get back to the editor and there you are. 4694 lines (the maximum allowed) at 80 characters each gives 375520 characters. Magic? No way. Just a bit of data compression.

If you fill memory with lines consisting of DIFFERENT characters, the limit is just over 283 lines (about 22670 bytes).

How is it done? Well, store some text. Unplug TI-WRITER plug in MINIMEM and off you go, strolling down memory lane.

TI-WRITER stores text from >A410 to >FD6B, aprox. 22876 bytes. (ED.ASSEMBLER uses from >A000 to >FD74, 23925 bytes.)

The program only allows use of the standard ASCII code, decimal codes 0 - 127 or HEX >00 to >7F for TEXT. Codes 128 to 255 or >80 to >FF, (they're not ASCII) are not allowed. If you just enter one word into one line of the editor, say 'ABCD' then look at memory starting at >A410 you will see that it is stored as follows :- (They've even got ME talking HEX now!)

```
>07 total # of bytes in record
>41 letter 'A'
>42 letter 'B'
>43 letter 'C'
>44 letter 'D'
>A0 which is >20 + >80 see below
>4C the decimal number 76
```

First we have a length byte >07 (which includes itself.) Next we have 'ABCD' which are the

characters in our record. Then comes the tricky bytes >A0,>4C. >A0 is really >20 but to show that it is REPEATED it is SHIFTED to the UPPER BANK of codes. If you prefer decimal then >A0 is 160 which comes from 32 (ASCII for SPACE) + 128 Get the idea?

Whenever a character is repeated it is indicated by a code offset by 128 followed by the number of repeats. In this case it is followed by >4C which every astute TIUPETEER knows instantly as decimal 76.

Why 76? Well, how many characters can you have in a line? 80 of course! You'll get an ELEPHANT stamped on to your wrist for that..... Now what happens when you take 'ABCD' (4 characters) away from 80? 76! Ahhhhhhhh! An elephant stamped on the other wrist..... What a smart idea, count how many unused characters there are at the end of each line and stick that information at the end. Is that a VARIABLE record or a FIXED record? Or a bit of both?

Let's try one more example. Purge memory and type in eighty letter 'Z's. Go back to MINIMEM and what do you find at >A410?

```
>03 three bytes to the record
>DA decimal 218, is greater than
127 so it must be a repeated
character, subtract 128, gives
90 which is ASCII for 'Z'
>50 decimal 80 which equals the
number of 'Z' repeats.....
```

And of course no >A0 at the end of this record because the line is full as a goog. Easy Huh?

OK smarty pants, now it's your turn. How are the following three lines stored :-

```
0001 AA bbbb CCCCCCCC dddddddddd
dddddd FFFFFFFFFFFFFFFFFFFFFFFF
FFFFFF TI-99/4A.!!!
0002
0003 Qquit c/r c/r p/h
```

(That's two carriage returns and a page header on the last line.)

What use is there in this information? Not all that much but don't you like to know how things are done? At least it will give you some idea of what wastes memory and what doesn't. You could separate text with lines of asterisks or other repeated symbols at no great memory cost.

SCENE AROUND

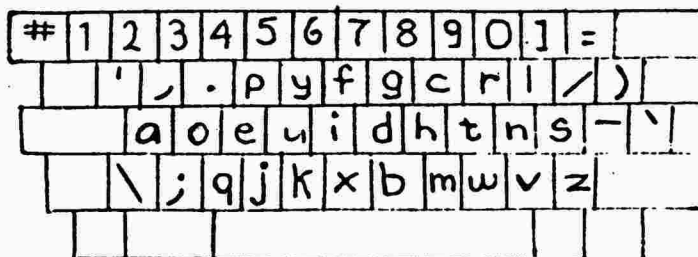
Looking back through my old TI magazines, it was interesting to re-read an article written by Peter Brooks (Tidings, 1982), 'Changing Times', which was an exercise to get us thinking about our future with the new Information Technology. Some of the possibilities he pointed out then are gradually becoming part of our daily lives: shopping by computer (Bradford and Gateshead councils have set these up already, but teletext advertising reaches almost everyone); all schools now have at least one computer (so the kids know what they look like!); we have electronic mail such as 'Prestel'; PC's are now so sophisticated we can do our own home publishing. In health care there have been developments in muscle stimulation to help disabled people walk, and in environmental controls, to help those who are still chair-bound. One or two people have a terminal so they can work at home, and beat a disabling disease.

And, slowly, we're losing the old 'work ethic' that was holding us back. Most of us now live with the 'fear of unemployment': even if you're working now, any number of factors beyond your control could alter that overnight. We're learning to live in a rapidly changing world, and our new awareness and sense of caring will monitor and modify these new developments so they are put to the best use. I'd like to make SCENE AROUND a regular page, to report some of the new ideas and technology that are emerging, and in another five years we can look back and see how we've progressed! I'll be very glad of help, most of you will be more 'in touch' than I am.

* * *

Under the Data Protection act, all companies who have any sort of personal information now have to publish it, and we are seeing the first registers in public libraries. There are some 77,000 records available at the moment, showing who has registered and what sort of information they hold. This is to be updated with another 55,000 records, and after November 11th you will be able to see exactly what they've got on you!

A new keyboard layout, Dvorak, to help touch typists go faster. For those of you who are not touch



The Dvorak keyboard

typists, there's a special layout with blank keys to allow thought transference (I lied, but if only!

HOLD your head up high straighten your back and say, "Yup, I'm not a sucker for crass commercialism." or "Oh, dear you must be one of those ingrates that accidently bought a, what was it?"

I got rid of my mentor by forcing hi, to see the DIMness of his ways.

"Look at it," he said "it's not even a real keyboard."

"True" I replied. " But it not only works, it keeps working."

"It's only 16K 'giggle,giggle'"

"True" I replied "but it's 16 real K of 16 bit with 22 bit stacks, not a phantom 64K with imaginary ROM and 8 stingey bits.

I got tired of the quips, "Hey, Commodore turkey, I admit all, why don't you bring your wonderful machine over and show me for instance how that marvellous sound generator works."

"SURE",his pride enveloping him in an aura.

He soon returned, I had greta difficulty controlling the giggles. I watched fidgetty as he gently placed his eggshell like box of tricks on the table. I couldn't help thinking how it looked like a sun bleached flattened wiener with keys. He hooked it up to my old TV just as if he knew what he was doing.

"I tell you what", I said with a sardonic smile, "Now that we've got both on the same table why don't we both program a tune?" I had hit the right spot, his face paled.

"Well, er well, OK," he stuttered.

"How about this one, it's short. 'GHOST RIDERS'?" There was no answer, he nodded, affirmatively. I could see his breath begin to quicken. I swear I could see his chest punding as his heart went into high gear.

Suddenly feeling calm and undauntable, I opened the music book with a flourish, then like Boris Brott (conductor), Spectacles at the end of my nose I opened my manual to page III-7. Work commenced. Internally I giggled continuously for the next 40 minutes, THEN "I've finished!" I exclaimed.

"WHAT?" gasped my wide eyed companion. "You couldn't be."

With confidence, poise, and grace I pressed "R" then "U" then "N". the room fell into a breathless hush (ENTER).

Well it was'nt exactly the Hamilton Philharmonic, and there were a couple of sharps where there should have been naturals. BUT I DID IT, and in only 40 minutes.

"There," I said when the music fell silent.

"Wasn't very good." he remarked.

"Well let's hear yours.!"

The Commodore user thought for a minute, "I've got a great Tree with Christmas music on tape."

"So What," I growled "I want to hear GHOST RIDERS." Reluctantlt he pressed the appropriate keys. "None of them fell off this time."

I hadn't laughed so much since the last Richard Prior move. After all that time all he got was :- "PING, PING, PONG, PING." Trying desperately to control my mirth I said "is that it?"

"Yes", he said weakly. The real problem is that the 64 is, in computer terms "prehistoric technology". Modern machines are very clever, with gret amounts of knowledge built in 'ROM'. The T.I. has over 70K of ROM with another 36K in your extended basic. The commodore has only 64K, end of message.

" You've got no control over your pitch, attack or decay on your T.I..". That's a good common line used by Commodore users. Believe me it's not true. The control of the voices on the T.I. is not as good as the Commodore, BUT it does not take 3 years in University to get something that sounds almost as good.

It takes a special person to program music into the 64, he or she must have a good understanding of SOUND, MUSIC and PROGRAMMING in machine language. On your old T.I. all you need is sheet music and page III-7 of your users guide.

So why do I love Commodore?

Easy, I run a business selling and repairing computrs. You cant make a living repairing the Maytag of the computer world, i.e. Texas Instruments. On the other hand Commodore ????

But don't go off half cocked, the Commodore 64 is an excellent computer, and if programmed correctly will perform extremely well. It's sound capabilities are slightly better than T.I.'s and that's all.

For a few extra bucks you can get some real sound. I put a reverb (home built), a moderate 30 Watt stereo amplifier, and a pair of decent speakers on my system.

Even the games are mind boggling when you wind up the power and blast those aliens. I can't imagine why there's a lot of houses up for sale in our neighborhood.

NOTE!.. the opinions expressed are those of the author and in no way are those of this club or it's newsletter editor.

From TIT-BITS Vol 4 No 2 Nov 1985
TI Users - Perth (Australia)



MULTIPLAN

From Vol 6 No 3 March 1987 Newsletter
of the Ottawa ti99/4A Users Group

TIMP Tips and Techniques

by

Steve Zimmerman

This time around. I'm going to talk (or write!) about a variety of small and not -so-small tips to help you avoid trouble with your spreadsheet models. First, if you choose to lock the formulas in a worksheet make an unlocked backup BEFORE YOU LOCK THEM! Locking the formulas in a worksheet is easy-unlocking them requires you to do it piece-by-piece! So, just in case you need to make changes in the future, keep your unlocked backup available! If someone else will be doing data entry on a complex worksheet, it IS a good idea to have them working with a locked copy- this avoids problems such as having someone enter a number or label in a cell which contains formulas or information you use elsewhere in the worksheet.

When building a worksheet, work on one area at a time. This allows you to enter numbers to check to see that each small area of the worksheet does what you want it to do. Using this technique, you can build up each individual area so that it works, and then link the areas to produce subtotals, grand totals, and the like. This is rather like programming in FORTH -in that you define and test a word, and then use that word in further definitions once you have tested it.

Deleting your numbers can help you see if an error condition results, and you can then correct the cause of the error condition in setting up your formulas.

Use relative references wherever possible in building a worksheet (I seldom use absolute references for anything!). The greatest advantage of relative references is that they allow you to easily copy formulas which will be used over and over. An absolute reference cannot be copied and used in another area of the worksheet without being edited-a time consuming process, and one which is prone to errors!

I use relative references for adding up columns of numbers--say, to get a daily total. Adding up $R[-1]C+R[-3]C+R[-5]C$ adds the number 1 row up, 3 rows up, and 5 rows up, in THIS column (and so, can be copied to any column and do the same thing!), and places the total here. A formula of $R13C2+R11C2+R9C2$, will add the values in those 3 cells and place the total in whatever cell it is in. If copied from column 2 to column 3, it will STILL add up the numbers IN COLUMN 2--NOT IN COLUMN 3! To get it to add up the numbers in column 3, 4, 5, or wherever it is copied to, it must be edited to correct the column references after it is copied. This is, to put it mildly, a pain!

The most accurate way to enter formulas in a spreadsheet is by pointing. Some spreadsheets don't allow this, but we are fortunate because multiplan does. In fact, in multiplan, we can point not only in entering but also in editing formulas. This is sometimes referred to as "wander mode". Using this method of building a formula is simple--and also has the advantage of creating relative references.

To use this method, begin by placing the cursor or cell pointer in the cell in which you want the formula to be placed and the end result to appear. Key either \vee or $=$ to begin entering a formula. Use the fctn> E, S, D, or X (the arrow keys) to move the cursor to the first cell you want to add, subtract or whatever. As you do, you will see a relative reference appear on the command line after the word VALUE:--if you move the cell pointer up 1 cell, it will show VALUE: $R[-1]C$. If this is the first cell you want to use, press the key for the operator (+, -, *, /) you need (in this case, let's say +). The + will appear after the cell reference on the command line and the cell pointer will drop back to the cell you

are building the formula in. To select the cell you want to add to the one just above, move the cell pointer again using the arrow keys. As you move it, you will see the relative reference on the command line change. When you have found the next cell, again, enter the next operator, and continue in this manner until the formula is complete. When you have pointed to the last cell to be in the formula, press enter>. The cell pointer will drop back to the cell you are working in, and the number created by your formula will appear in that cell. In the lower left-hand corner of your screen, you will see the coordinates of the cell you are in, and the first 16 positions of the formula you have created (or the entire formula, if it is less than 16 positions long).

Using this method, you can easily enter long formulas without trying to remember cell coordinates and without taking the time (and memory overhead!) involved in naming cells. Formulas that operate in one column or row can be copied across or down and will work properly in any row or column.

When building a model this way, enter your row or column labels first, then enter data (sample data, if possible), and last, point to each data item to create your formula. Once you have the formula set up, blank the cells with your sample data items, let the sheet recalculate (or tell it to recalculate, if you turn off the automatic recalculations as I always do), and check for errors, such as #DIV/0! (meaning that you're trying to divide by 0). Save your backup copy when your formulas are done, then lock the formulas in your working copy (if you want to), and enter your data. This will help you in creating error-free models.

Next month, we'll go over some (potential) problems that can lead to your having different numbers appear in your models than you are expecting! Until then, have fun with multiplan!



The Ottawa T.I. 99/4A Users' Group



THE CLASSIFIED ADVERTISEMENTS PAGE

* * *

MINIMEM Conversion to Rechargeable Battery. Send Minimem and Crossed cheque £7.50, To: N.J. Petry, Tensal Technology, 15 PENRICE CLOSE, WORLE, W.S.M., AVON, BS22 9AH.



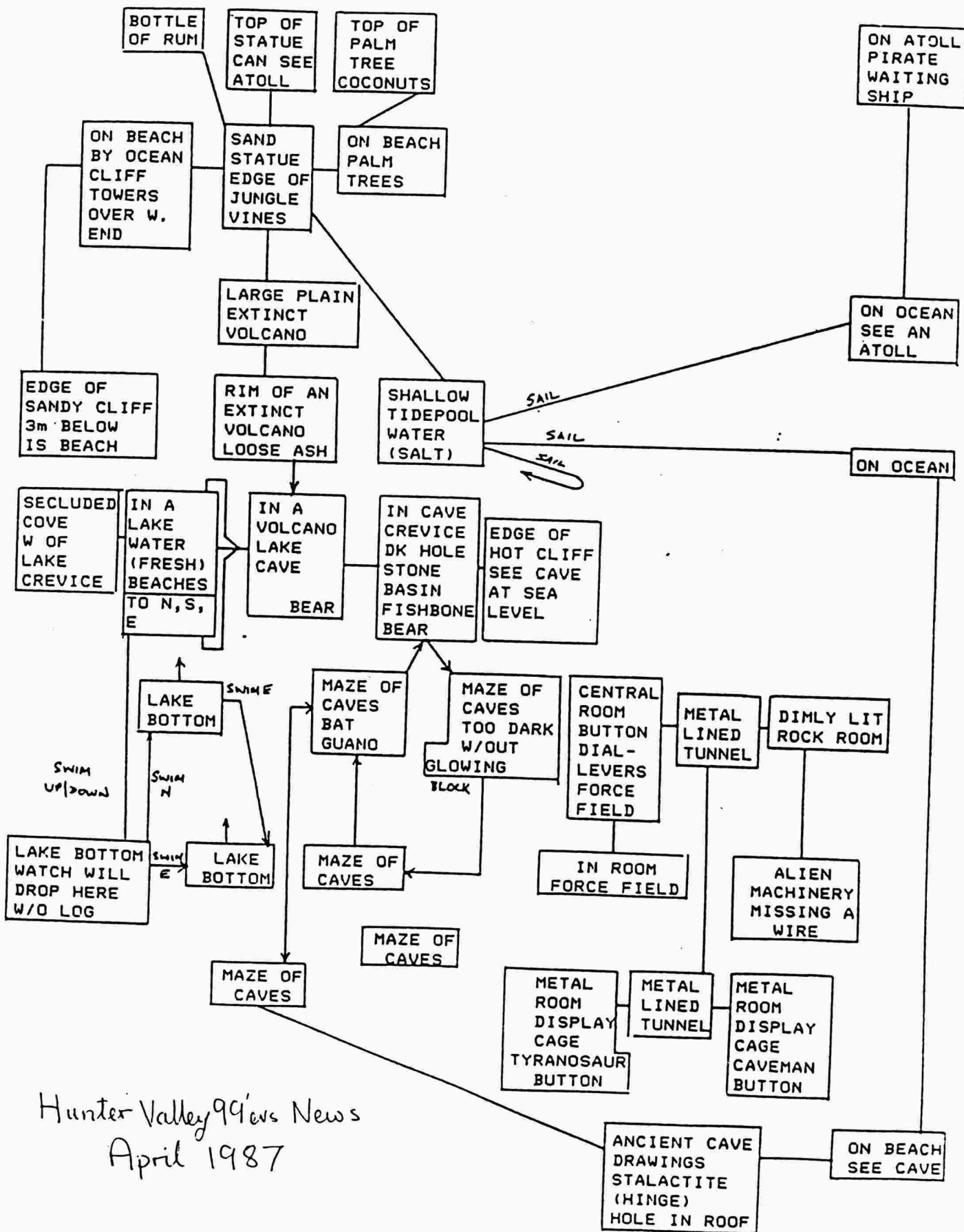
USE THIS SPACE - SEND US YOUR FOR SALE/WANTED ADS
for inclusion in the next issue of TI*MES

IS DOWN TO THIS !

Is everybody on holiday?Is there anybody out there?

SAVAGE ISLAND

PART 1



This article is a follow up to the series written by
Stephen Shaw and published in TIMES Issues 6 and 7
and comes to us from the LEHIGH 99'ER COMPUTER
GROUP magazine Vol. 111 no. 3 - March 1985 written by
Frederick Hawkins.

XB+32K
+DISK

100 DATA 03AC1AB010
110 DATA 03860D9210
120 DATA 08908F08AD11C315D01
0
130 DATA 028A0A10
140 DATA 07D291BF0CB238F0720
150 DATA 03B0C31510
160 DATA 028A0A10
170 DATA 05BF8F08CC1A10
180 DATA 02860D10
190 DATA 05B1C3158A0A10
200 DATA 028F0710
210 DATA 03BF0CB2310
220 DATA 028F0810
230 DATA 058A0AB2C31510
240 DATA 02860D10
250 DATA 049FBFCC1A10
260 DATA 03860D9110
270 DATA 05B1C3158F0810
280 DATA 038A0A9010
290 DATA 058F07BFC41920
300 DATA 04B1CE0F9110
310 DATA 028A0A10
320 DATA 05DF800AAD1710
330 DATA 028E0B10
340 DATA 05D1A014800A10
350 DATA 03908F0710
360 DATA 04DFAC1F9110
370 DATA 02800A10
380 DATA 05D28A0AA01410
390 DATA 028E0B11
400 DATA 039FBFDF0101B20F
410 DATA 07B08107C00A91D210
420 DATA 03AD11D20F019401
430 DATA 0290D10F0295D301
440 DATA 06AB23D1918F0720
450 DATA 0295D2010491D0AD111
0
460 DATA 049FA014DA0101D10F
470 DATA 03C20ED010
480 DATA 05D1CE0BAD1110
490 DATA 02C00A10
500 DATA 04DFB1AB2310
510 DATA 01D210
520 DATA 07AD11B08F0891DF10
530 DATA 028F0710
540 DATA 039FA01410
550 DATA 07B08107C00A91D210
560 DATA 03AD11D20F019401
570 DATA 0290D10F0295D301
580 DATA 06AB23D1918F0720
590 DATA 0295D2010491D0AD110
600 DATA 06DF8A0AAC1F9010
610 DATA 02860D10
620 DATA 078E0BAD11C315D110
630 DATA 03918A0A10
640 DATA 04BF0CB239210
650 DATA 03918E0B10
660 DATA 03DF8A0A10
670 DATA 02800A10
680 DATA 03AC1AB010
690 DATA 03860D9210
700 DATA 08908F08AD11C315D01
0
710 DATA 028A0A10
720 DATA 07D291BF0CB238F0720
730 DATA 03B0C31510
740 DATA 028A0A10
750 DATA 05BF8F08CC1A10
760 DATA 02860D10
770 DATA 05B1C3158A0A10
780 DATA 028F0710
790 DATA 03BF0CB2310
800 DATA 028F0810
810 DATA 058A0AB2C31510
820 DATA 02860D10
830 DATA 06CC1A8B06BFD010
840 DATA 01980101900F
850 DATA 03CD179110
860 DATA 01990101900F
870 DATA 0698A007B2C31501019
01F
880 DATA 05D1CC1A98B7010291B
11F
890 DATA 0598AF07C0140101900
F
900 DATA 03800A8010
910 DATA 04CC1A8F0710
920 DATA 028B0610
930 DATA 0391C01410
940 DATA 02810710
950 DATA 03DF8B0610
960 DATA 02810710
970 DATA 098B06A40BCE2190B0D
020
980 DATA 039FBFDF010390B0D03
F
990 DATA 099FBFDF8F08AA0ACD1
120
1000 DATA 0390B0D020
1010 DATA 039FBFDF010390B0D0
3F
1020 DATA 03DF8FBF20
1030 DATA 01D210
1040 DATA 019110
1050 DATA 05B192DFAB2310
1060 DATA 02870810
1070 DATA 0790B0D1A00A8F0720
1080 DATA 06A1078008CB2320
1090 DATA 07D0CC1A8B06AA0A22
1100 DATA 039FBFDF20
1110 DATA 0390B0D020
1120 DATA 04FF9FBFDF2E0
1130 DATA 000400,
1140 CALL INIT
1145 !"fly, you blackbirds"
from Brainard's Ragtime
Collection (1899)
1146 ! arranged by
DENES AGAY
(from The Joy of Ragtime
selected & edited by
Denes Agay



```

1147 !Yorktown Music Press,
Inc)
1148 ! pgm by Fred Hawkins
1150 FOR VOICE=0 TO 3 :: CAL
L XOFF(VOICE):: NEXT VOICE
1160 X=32 :: CALL DISABLE
1170 CALL LOAD(-31796,4,0,1)
!set sound list and trigger
1180 CALL LOAD(-31747,1)!set
sound source pointer
1190 READ A$ :: PRINT A$ ::
B$=B$&A$
1200 IF LEN(B$)>64 THEN GOSU
B 1240
1210 IF A$>"" THEN 1190
1220 IF LEN(B$)>1 THEN PRINT
LEN(B$);B$ :: GOSUB 1240
1230 GOTO 1260
1240 C$=SEG$(B$,1,64):: B$=S
EG$(B$,65,64):: PRINT : :B$:
:

```

```

1250 PRINT "LOADING";X;LEN(
C$):: CALL CHAR(X,C$):: X=X+
4 :: RETURN
1260 ACCEPT AT(4,4):C$ :: IF
LEN(B$)>64 THEN CALL CHAR(1
32,SEG$(B$,65,255))
1270 CALL ENABLE
1280 PRINT "PLAY"
1290 FOR A=1 TO 12 :: CALL C
OLOR(A,3,13):: NEXT A :: CAL
L MAGNIFY(4)
1300 CALL SPRITE(#1,32,16,40
,50,1,3)
1310 FOR A=0 TO 80 :: CALL P
ATERN(#1,32+A):: CALL COLOR
(#1,13*RND+3):: NEXT A :: GO
TO 1310
1320 SUB ENABLE
1330 CALL LOAD(-31806,0):: S
UBEND
1340 SUB DISABLE
1350 CALL LOAD(-31806,32)::
SUBEND
1360 SUB XOFF(VOICE)
1370 A=(VOICE-1)*32+159
1380 CALL LOAD(-31744,A):: S
UBEND

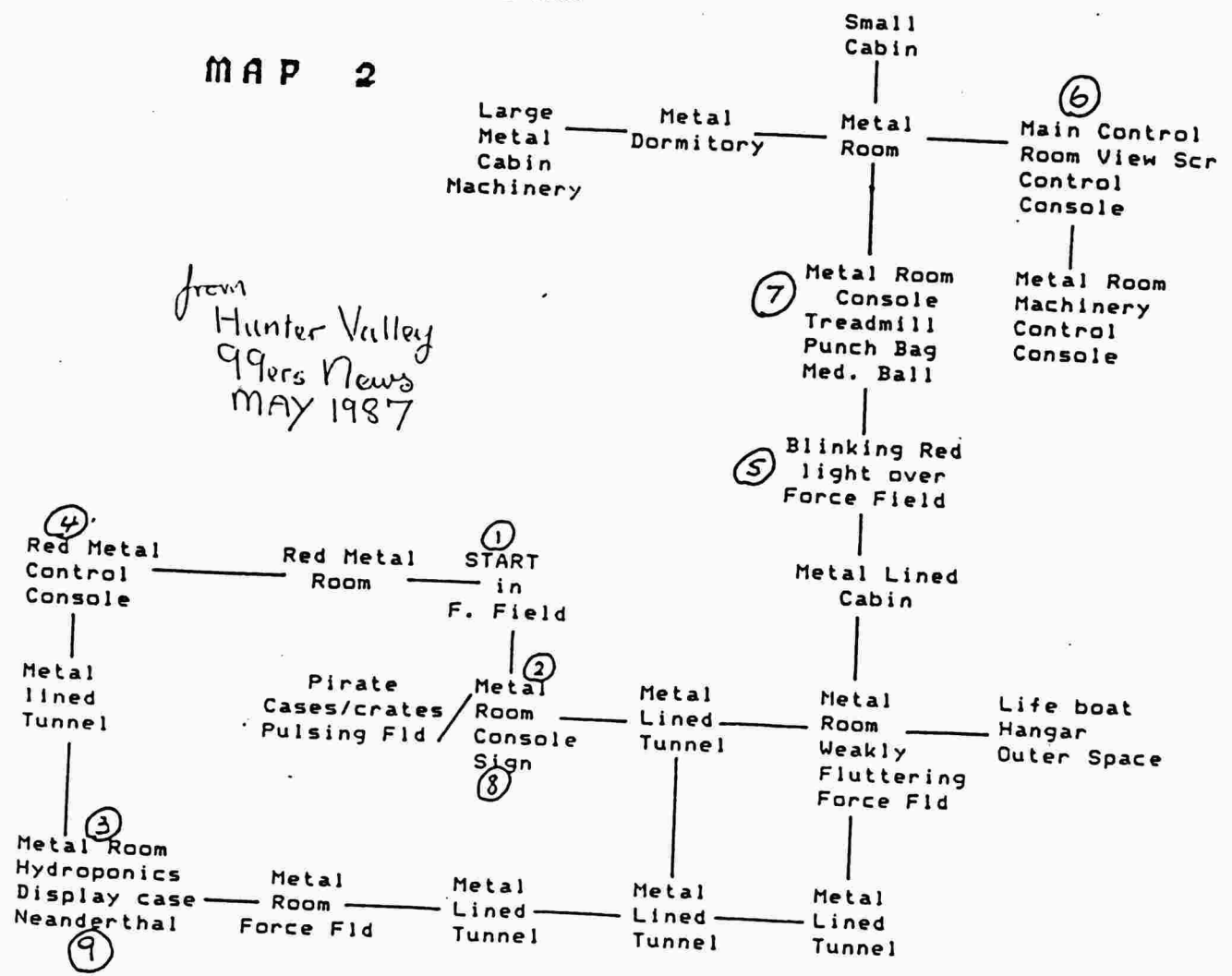
```

from Sydney News Digest
Oct '85



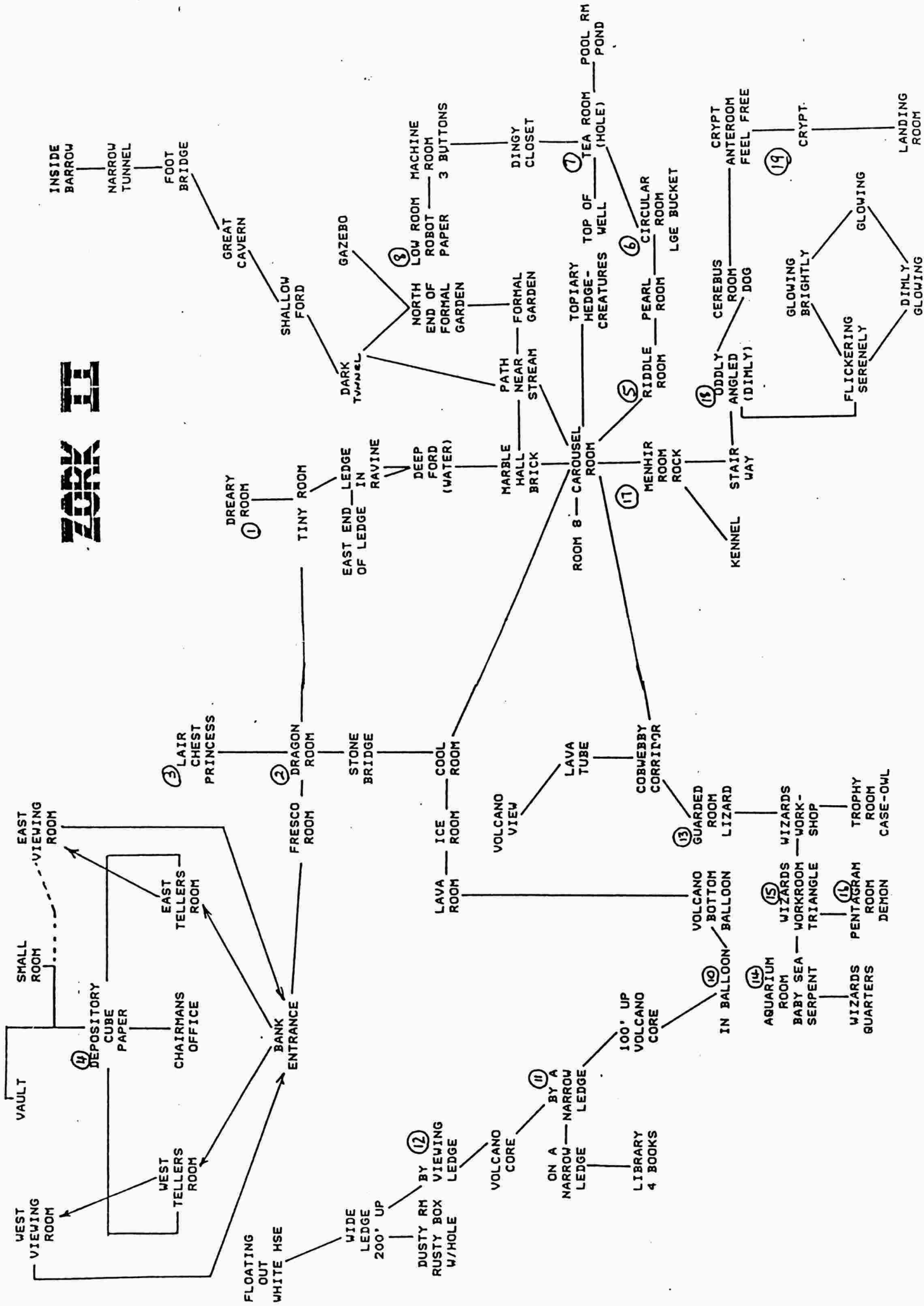
SAVAGE ISLAND

MAP 2



from Hunter Valley
99ers News
MAY 1987

ZORK II



This page is blank

of the welfare state. It is important to note that the effects of the welfare state on the labor market are not the same as the effects of the welfare state on the economy as a whole.

The effects of the welfare state on the labor market are analyzed in this paper. The paper is organized as follows. Section 2 discusses the effects of the welfare state on the labor market. Section 3 discusses the effects of the welfare state on the economy as a whole. Section 4 discusses the effects of the welfare state on the labor market in the long run. Section 5 concludes.

2. Labor market

The labor market is analyzed in this section. The labor market is divided into two parts: the labor market for high-skilled workers and the labor market for low-skilled workers.

The labor market for high-skilled workers is analyzed in this section. The labor market for high-skilled workers is divided into two parts: the labor market for high-skilled workers in the private sector and the labor market for high-skilled workers in the public sector.

The labor market for low-skilled workers is analyzed in this section. The labor market for low-skilled workers is divided into two parts: the labor market for low-skilled workers in the private sector and the labor market for low-skilled workers in the public sector.

The labor market for high-skilled workers in the private sector is analyzed in this section. The labor market for high-skilled workers in the private sector is divided into two parts: the labor market for high-skilled workers in the private sector in the short run and the labor market for high-skilled workers in the private sector in the long run.

The labor market for high-skilled workers in the public sector is analyzed in this section. The labor market for high-skilled workers in the public sector is divided into two parts: the labor market for high-skilled workers in the public sector in the short run and the labor market for high-skilled workers in the public sector in the long run.

The labor market for low-skilled workers in the private sector is analyzed in this section. The labor market for low-skilled workers in the private sector is divided into two parts: the labor market for low-skilled workers in the private sector in the short run and the labor market for low-skilled workers in the private sector in the long run.

The labor market for low-skilled workers in the public sector is analyzed in this section. The labor market for low-skilled workers in the public sector is divided into two parts: the labor market for low-skilled workers in the public sector in the short run and the labor market for low-skilled workers in the public sector in the long run.

The labor market for high-skilled workers in the private sector in the short run is analyzed in this section. The labor market for high-skilled workers in the private sector in the short run is divided into two parts: the labor market for high-skilled workers in the private sector in the short run in the short run and the labor market for high-skilled workers in the private sector in the short run in the long run.

The labor market for high-skilled workers in the public sector in the short run is analyzed in this section. The labor market for high-skilled workers in the public sector in the short run is divided into two parts: the labor market for high-skilled workers in the public sector in the short run in the short run and the labor market for high-skilled workers in the public sector in the short run in the long run.

The labor market for low-skilled workers in the private sector in the short run is analyzed in this section. The labor market for low-skilled workers in the private sector in the short run is divided into two parts: the labor market for low-skilled workers in the private sector in the short run in the short run and the labor market for low-skilled workers in the private sector in the short run in the long run.

The labor market for low-skilled workers in the public sector in the short run is analyzed in this section. The labor market for low-skilled workers in the public sector in the short run is divided into two parts: the labor market for low-skilled workers in the public sector in the short run in the short run and the labor market for low-skilled workers in the public sector in the short run in the long run.

The labor market for high-skilled workers in the private sector in the long run is analyzed in this section. The labor market for high-skilled workers in the private sector in the long run is divided into two parts: the labor market for high-skilled workers in the private sector in the long run in the short run and the labor market for high-skilled workers in the private sector in the long run in the long run.

The labor market for high-skilled workers in the public sector in the long run is analyzed in this section. The labor market for high-skilled workers in the public sector in the long run is divided into two parts: the labor market for high-skilled workers in the public sector in the long run in the short run and the labor market for high-skilled workers in the public sector in the long run in the long run.

The labor market for low-skilled workers in the private sector in the long run is analyzed in this section. The labor market for low-skilled workers in the private sector in the long run is divided into two parts: the labor market for low-skilled workers in the private sector in the long run in the short run and the labor market for low-skilled workers in the private sector in the long run in the long run.

The labor market for low-skilled workers in the public sector in the long run is analyzed in this section. The labor market for low-skilled workers in the public sector in the long run is divided into two parts: the labor market for low-skilled workers in the public sector in the long run in the short run and the labor market for low-skilled workers in the public sector in the long run in the long run.