HUNTER VALLEY 99ERS USERS GROUP

HOME COMPUTER NEWSLETTER
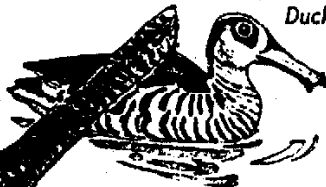
Hunter Valley 99ers USER GROUP

Red-capped
Parrot
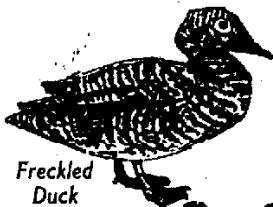
Pink-eared
Duck

Freckled
Duck

Splendid
Wren

Noisy Scrub-bird

JUNE 1990

YOUR COMMITTEE

PRESIDENT
Peter Smith
8 Glebe St.
East Maitland 2322
Phone 336164 V/tl 493261640

VICE PRESIDENT
John Paton
1 Parlen Close
Rutherford 2320
Phone 326014 V/tl 493260140

SECRETARY
Brian Woods
9 Thirlmere Pde.
Tarro 2322
Phone 662307 V/tl 493223070

TREASURER
Noel Cavanagh
378 Morpeth Rd.
Morpeth 2321
Phone 333764

SOFTWARE LIBRARIAN
Stewart Bradley
14 Hughes St.
Birmingham Gardens 2287
Phone 513246

EDITOR
Allen Wright
77 Andrew Rd.
Valentine 2280
Phone 468120

PURCHASING/HARDWARE CO-ORDINATOR
Alan Franks
822 Pacific Highway
Marks Point 2280
Phone 459120

SOCIAL SECRETARY
Robert (Bob) MacClure
75 Deborah St.
Kotara South
Phone 437431

PUBLICATIONS LIBRARIAN
Ken Lynch
9 Hall St.
Edgeworth 2285
Phone 585983

COMMITTEE MEMBERS
Don Dorrington
36 Nelson St.
Barnsley 2301
Phone 531228

Tim Watkins
36 The Ridgeway
Bolton Point 2283
Phone 592836

CONTRIBUTIONS

Members and non members are invited to contribute articles for publication in HV99 News.

Any copy intended for publication may be typed, hand written, or submitted on disc media as files suitable for use with TI WRITER. (ie. DIS/FIX 80 or DIS/VAR 80). A suitable Public Domain word processor program will be supplied if required by the club librarian.

Please include along with your article sufficient information to enable the file to be read by the Editor eg. File name etc. The preferred format is 75 columns and page length 66 lines, right justified.

All articles printed in HV99 NEWS (unless notified otherwise) are considered Public Domain. Other user groups wishing to reproduce material from HV99 News may feel free to do so so long as the source and author are recognised.

Article for publication can be submitted to the Editor, ALL other club related correspondence should be addressed to the Secretary.

DISCLAIMER

The HV99 News is the official newsletter of the HUNTER VALLEY NINETY NINE USER GROUP.

Whilst every effort is made to ensure the correctness and accuracy of the information contained therein, be it of general, technical, or programming nature, no responsibility can be accepted by HV99 News as a result of applying such information.

The views expressed in the articles in this publication are the views of the author/s and are not necessarily the views of the Committee, Editor or members.

TEXAS INSTRUMENTS trademarks, names and logos are all copyright to TEXAS INSTRUMENTS.

from President Peter's Quill.

Well that time of the year has come around again..   Tax and club executive elections.

As you all know a lot of discussion has centred around this club's makeup next year and in view of the lastest information we can look forward to only a recommitment of at the most 50 members, most of whom are remote from this area. This means that the structure of the club needs review and this is exactly what happened at the last committee meeting. The committee moved that the size of the executive be cut to the minimum..

PRESIDENT or COORDINATOR or ORGANISER(whatever you like).
TREASURER someone has to look after the money.
SECRETARY someone has to write the letters and keep notes.

MEMBERS
Ordinary members are asked to to assist their club in whatever way he/she can . We have had a number of people who have mention that they will be prepared to assist by helping with newsletter publication and library (both software and publications) duties.

It is anticipated that the club will function just as well as ever as just about all local active members are at present on the committee, and they will turnup at committee meetings anyhow. We simply need to nominate only a minimum number of positions.

Please assist your club to continue to opperate for your benefit this coming year by paying your fees promptly and assisting in whatever ways you can. The newsletter certainly needs your input.. let's show what that's there's life in the old machine yet.

The updated software library should be of great assistance in renewing interest in the club.

Lots of programs which noone has any idea about are in our posession, please try some that interest you and write a little about it so that we can all benefit from your experience. (Whether it be good or bad review.)

We also have lots of useful programs which can add some extra life to your machine... try them.. they may open a whole new vista for you.

Thanks to all those hardworking executives who have served the club well this last 12 months.. Well done fellows. I'm sure you wont rest on your laurels though and I know you will be involved nex year. Thanks go also to all our members who have supported us so well with contributions to our magazine and in many other ways.
Peter.

Brian WOODS REPORTS from the
SECRETARY'S DESK

US VISIT

Recently my family & I were able to take a holiday on the West Coast of the US. Before going I was armed with phone numbers & people to contact in the areas I was to visit. One such contact was Earl Raguse of the Brea 99ers Users Group.

After settling in at Anaheim I gave him a call and was invited to attend the monthly meeting of the group. After a little confusion (earl was early & I was late!) we managed to get to the meeting. Among their members are Ken Hamai (who couldn't make it to the meeting), Bob August, who is a prolific writer/programmer whose articles appear regularly in their newsletter, and of course, Newt Armstrong.

After a welcome by the Vice President, Kerry Grissom, I was invited to say a few words on TI'ing `Down Under'. Following this was general discussion and a demo of a new program written by Earl. The evening concluded with informal talking over a cup of coffee (you haven't drunk coffee till you have drunk it in America — it's beaut!).

All in all, the meetings were run along much the same as our own, but it was a great experience meeting some of those people who you only read about. Thanks for your hospitality Kerry & members, & a special thanks to Earl for providing the transport.

THE TI & THE McGOVERN'S GET A MENTION!

Australian Personal Computer magazine published a "Commemorative 10th Birthday Supplement" in May, and one article dealt with the last 10 years in computing. No account of the past decade could be written without some mention of the TI, and sure enough, it was covered, albeit quoting the wrong model number!

"It was a bleak day for many in October 1983 when Texas Instruments decided that it had had it with the home computer market and dumped the TI-99/4. The company had sold mega-quantities of the machine but had forgotten a golden rule of business... you need to sell your product for more money than it costs to make!"

"More than six years later, the computer is neither gone nor forgotten. There are user groups galore all around the world and a year or so ago I discovered that two Australians were doing their bit to keep users happy and productive. University of Newcastle physics professor, Tony McGovern, and his 16 year old son Will, created Funnel Web, a word processor named after the infamous spider. At the time, one enthusiastic US user told me that the latest version, Funnel Web 4.0, was the most single important programing effort that's been produced for the TI. Another said, "Funnel Web is the king of the Ti word processors".

"The early days of the personal computing business attracted an oddball group of devotees — I'll never forget being crushed at shows in London which were crammed with people who owned the tiny Sinclair ZX computers — and because of their efforts, the relics of those bygone times are being preserved."

NEW SPELLING CHECKER

From the April issue of the Brisbane User Group newsletter comes the following...

Asgard Software have just released a companion for the word processors. It is called SPELL IT and is the fastest and most complete spelling checker that has been released for this computer. It is claimed to be compatible with all files created by TI word processors (DV80). It will make corrections directly in the text, display the mis-spelt word in context, lets you add words to the user dictionary, has no limit on the size of the file that can be checked and allows you to scan the dictionary if you are not certain of the correct spelling of a word.

The Smartcheck (TM) algorithm means that it does not have to individually check for plural & possessive versions of the words in the dictionary. The floppy disk version has a dictionary of over 25,000 words and the hard disk version has over 200,000 words. It is supplied un-protected and is RAMdisk compatible and can be configured for 80 columns. It is compatible with both TI 99/4A & Geneve.

The cost is:
DSDD - $US19.95   SSSD - $US24.95   HFDC - $US34.95   Add $US6.00 airmail

TETRIS - a game review

From the April issue of the Sydney User Group comes this review...

Tetris is a computer program from the soviet Union, now converted for most computers and even released as a coin-op machine, it is THAT good. Very simple in concept, it can take a moment or two to realize in play what you are meant to do - different shapes fall from above, and by rotating them, and moving them left and right, you have to do all you can to pack them so tightly that complete rows with no holes are made up - when you do the whole stack moves down as that row disappears.

The first TI version was TI TRIS. the version that follows is Tetris and comes from the September 1989 issue of MICROpendium. As with all programs with DATA statements, take very great care in keying it in! The printout is from a program which has been up and running correctly, and can therefore be guaranteed free from bugs!

ROCK RUNNER

From the March issue of TIdbits. the newsletter of the Mid South Users Group, comes this game review by Gary Cox...

New from Asgard Software of PO Box 10306 Rockville, MD 20849 USA is Rock Runner, a 15 level arcade game similar to Boulder Dash for Atari computers. The game is said to take advantage of heretofore unused graphics modes of the TMS9918A video processor found in the TI 99/4A. The game operates under the Editor/Assembler module. 32K and a disk system and is played with a joystick. The object is to move the figure around the playing area (larger than the screen, which windows to show the area the character is in) and collect a specified number of diamonds within a given time. Complicating matters are rocks or boulders that can fall on the player and at the higher levels, monsters that are interested in lowering the players life expectancy. At higher levels the player will have to set traps, blow up walls with bombs and perform acrobatic feats with the joystick to obtain the diamonds. The game retails for $US12.95 plus postage.

"THE ULTIMATE GRAPHICS UTILITY"

That is the headline in the February issue of MICROpendium in an advertisement for a newly released collection of graphics. The ad continues...

TIPS is the ultimate graphics utility for the 99/4A. Now with one low priced utility you get a program that allows you to create your own custom greeting cards, banners, flyers, posters and much more. Best of all is the TIPS library of over 2000 graphics including every type subject matter and borders and margins. This fantastic freeware program contains the largest printshop library that has ever been available for the 4A. Not only can you print out this library in the sizes you need, you can also convert all the artwork to the TI-Artist format and all other popular 4A formats. In addition to the TIPS program disk at only $4.95, the giant TIPS library of over 2000 graphic files comes on ten two sided disks which can be purchased separately at only $4.95 each or in a complete package for a special price of only $39.95 with the TIPS program disk, a $4.95 value, included at no additional charge.

The package can be purchased from Tex-Comp in the US.

FOR YOUR INFORMATION
The following tid-bits come from the October 89 issue of Net99er News. The were compiled by the editor, Barbara Massey, from information received from various sources...

1) Did you know that IBM invented the small non-removable hard drive named Winchester, apparently because the drives code number matched the model of a popular Winchester rifle?

2) Did you know the disk inside a hard drive is referred to as a platter and the axle they are mounted on is called the spindle?

3) Did you know that DMA means Direct Memory Access, not Dump memory Alright?

4) Did you know that on many hard drives that the heads fly over the platters at a distance of 1/100,000 of an inch?

5) If your thinking of adding a second drive to your system you might consider this. Most of the older drives will work fine, but the newer drives called AT drives won't work on your TI machines. The reason for this is that the older drives, referred to as XT drives or 360K drives run at a rate of 300 RPM, and the newer AT drives operate at 360 RPM, plus the access time is less due to faster head steps.

BOB CARMANY REPORTS
It was interesting to read the article Bob wrote for the Guilford User Group after his recent trip out here. He certainly gave us all a big wrap-up, and for that we thank him. It appears that he had almost as good a time being entertained as we had entertaining him! It was great having you out here Bob and I am pleased to hear that next time you will bring your wife Sandy with you.

TI-BASE
At the April meeting some interest was shown in this rather complex database by some of our members, so for those who are in the process of experimenting with it, here are a couple of tips taken from exchange newsletters.

The first is from Rick Lilley writing in the March issue of the newsletter of the Hamilton, Ontario user group...

Here's a tip some of you might find useful! If you modify your "SETUP" command file slightly, you won't have to worry about, or include in any of your command files, your printer calls or parameters. In my case, my printer being a Star Micronics NX10, I typed "SET DATDISK=DSK1." to change my data disk setting from my normal setting of DSK2. Then I typed "MODIFY COMMAND SETUP". This has the effect of putting on the screen, for you to modify, the SETUP command file that TI-Base auto boots when you fire it up! All you have to do then is

type "PRINTER NX10", or whatever your printer is, and press FCTN 8. Now type "DO SETUP", & TI-Base will run SETUP again, and load your printer parameters from the PRINTER database. It will do this automatically from now on. It slows the loading process slightly, but you will never again have to be concerned about including a printer command in any of your files or getting an error warning on screen when you tell TI-Base to print something directly from the "DOT PROMPT" or immediate command mode!

Something I just discovered! When you're in the editor of TI-Base and you want to get to the first line of your command file in a hurry, just press FCTN 6. ZAP! You are now on the first line. I don't remember seeing that one in the manual!

The second tip comes from the PUG Peripheral, and first appeared in the LA99ers Users Group newsletter. The author is Chick De Marti.

Leave it up to a lazy man to get something done the easy way. This command file is called DO IT.

Here is the premis. You decide to write a program file to print a modified database, and you name it PMINLST... so you type DO MC (shortcut #1). When TI-Base asks you for the name you type PMINLST. Great! But as you continue to write your program, from time to time you test it, find it needs correction, and start the DO MC... wait then enter PMINLST. After an hour of this, I decided to do something about it. I wrote the program file "DO IT"!

Type DO MC
then IT in answer to what name?
at the blank screen I typed:
MODIFY COMMAND PMINLST
& FCTN 8 to save.

Now while I am working on this particular program and I need to make a correction, I type;

DO IT

One last note on the DO IT program. Whenever you decide to work on another program, go into the DO IT program and change PMINLST to the program you want to work on!

IT'S THAT TIME OF YEAR
June is that time of year when the current Committee starts to wind down in preparation for the taking over of the group by new blood. If you think you have something to offer the group, why not nominate for a position on the Committee? I can guarantee that you will enjoy the experience (after all, most of the current Committee have been on it before, so it must be good!!), and who knows, your ideas may bring more members into the group.

This is also the time of year when membership dues become payable. Please assist by paying yours as soon as possible. A letter was sent out to all our Australian members seeking their intentions regarding re-joining, and it is gratifying to note that most of our `out of towners' have already, or intent to, rejoin. One of the most requested things is another `Great Mailout' should be undertaken as soon as possible. I am pleased to announce that that request is in hand, and a mailout should be underway early August to all PAID UP out of towners.

One of the many things discussed by the Committee at their final meeting was suggested formats for monthly meetings. It was decided that the new Committee should offer more `hands-on' computing with a minimum of formalities at the meetings. Another suggestion was that the vast majority of the demos etc at the meetings should be aimed at the `average' user, ie Disk Drives, Funnelweb 40 column etc. All

members are canvassed for their suggestions, criticisms and other ideas.

RISK Vs ANXIETY

From the PUG Peripheral, April 90, comes this thought-provoking article...

"One of the problems with our country is that we don't have the ability to distinguish between risk and anxiety, entirely different things. David Padwell, chairman of Hydogentics, stated it very well. He said, for instance, we know the Surgeon General tells us 150,000 people die every year from smoking cigarettes, but we're not afraid of cigarettes; 50,000 people die in automobiles, but we're not afraid of automobiles. What we're afraid of is sharks. And in reality — he went on to say — the Medical Statistics Board doesn't even keep a record — they can tell you how many people die of bee stings every year, but not how many die of shark bites. The Navy guesses somewhere around two a year. But the fact is, if you went to a beach and shouted Shark! Shark! everyone would rush out of the water and get in his car and light a cigarette!"

THE END

Herbert Hoover may have had our Committee elections in mind when he said:

"In the great mass of our people there are plenty individuals of intelligence from among whom leadership can be recruited."

TEXAS INSTRUMENTS
Chip chat
©1981 TEXAS INSTRUMENTS
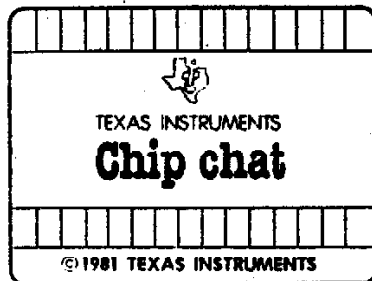
```
*************************************
*       CALL KEY SUBROUTINES       *
*************************************
```
When you specify a "key-unit" of 1 or 2(for split keyboard scan) in the CALL KEY subroutine, the computer does not return a true 0(zero) in the return-variable when you press X on keyboard 1 or M on keyboard 2. Therefore, if your programme checks to see whether the return-variable equals zero, the computer returns an answer as false.
The following example illustrates another way to write your program.

```
100 CALL KEY (1,A,B)
110 IF B<>1 THEN 100
120 IF A+1=1 THEN 200
    .
    .
    .
```

NOTE that, if line 120 read "IF A=0 THEN 200," the programme would not work properly.

```
*************************************
*    PRIME    NUMBER    GENERATOR   *
*************************************
```
Here is a program for finding prime numbers that works. It runs in XB but is a bit slow finding the larger numbers but is O.K. for smaller numbers (<~50) It works this way (in thoery anyway, the program is a bit different), it takes a list of numbers then looks through and takes out any that are divisible by 2. After it has done this it repeats for 3,4,5 etc. until it reaches the end of the list. The numbers that are left are the primes.

```
C.MAC
100 ! SIEVE OF ERATOTHENES
110 ! C.MAC
120 ! JUNE '85
130 ! FOR H'V 99'ERS NEWS
140 CALL CLEAR
150 PRINT "SIEVE OF ERATOSTHENES-"
160 PRINT "FOR FINDING PRIME NUMBERS"
170 INPUT "STARTING AT ? ":Q
180 INPUT "GOING TO ? ":A
190 DIM B(1000)
200 FOR S=2 TO A
210 B(S)=S
220 NEXT S
230 FOR K=2 TO A
240 FOR T=K TO A
250 IF B(T)/K=1 THEN 270
260 IF B(T)/K=INT(B(T)/K)THEN B(T)=0
270 NEXT T
280 NEXT K
290 FOR U=Q TO A
300 IF B(U)<>0 THEN PRINT B(U)
310 NEXT U
```

By Bob Carmany

Some years ago, Miller's Graphics put out a newsletter called "The Smart
Programmer". In the beginning, it was very interesting and it contained
some excellent short programs. I have included this one because it will
convert a number in one number base to another. In fact, it will give you
decimal, hexidecimal, and binary equivalents of a number in any of the
three bases. Although it is written in XB, it is still reasonably fast.
If you wish, you could add a routine to dump the results to an appropriate
printer.

```
100 ON WARNING NEXT :: CALL CLEAR :: H$="0123456789ABCDEF" :: PRINT
"DEPRESS YOUR ALPHA LOCK KEY": :" PRESS LETTER FOR INPUT BASE": :

110 PRINT : :"D=DEC # H=HEX # B=BIN #": : :: CALL SOUND(80,660,6)

120 CALL KEY(0,K,S):: IF S<1 THEN 120 ELSE ON POS("DHB",CHR$(K),1)+1 GOTO
110,130,140,150

130 INPUT "DEC #=":DEC :: IF DEC<-32768 OR DEC>65535 THEN 130 ELSE
A,DEC=INT(DEC-65536*(DEC<0)):: GOSUB 200 :: GOSUB 220 :: GOTO 160

140 PRINT "HEX #=" :: ACCEPT AT(23,7)BEEP SIZE(4)VALIDATE(H$):HEX$ :: GOSUB
180 :: GOSUB 200 :: GOTO 160

150 PRINT "BIN #=" :: ACCEPT AT(23,7)BEEP SIZE(16)VALIDATE("10") :BIN$ ::
GOSUB 90 :: GOSUB 220 :: GOSUB 210

160 A=INT(DEC/256):: PRINT :"D=";DEC;TAB(12);A;DEC-A*256 :: IF DEC>32767
THEN PRINT " ";DEC-65536

170 PRINT "H= ";HEX$:"B=";SEG$(BIN$,1,8)&&" "&&SEG$(BIN$,9,8) ::
HEX$,BIN$="" :: ,DEC=0 :: GOTO 110

180 HEX$=SEG$("0000",1,4-LEN(HEX$))&&HEX$ :: FOR I=1 TO 4 :: A,DEC=
DEC+POS(H$,SEG$(HEX$,I,1),1)-1)*16 ^(4-I):: NEXT I :: RETURN

190 FOR I=1 TO LEN(BIN$):: DEC=DEC-2^(I-1)*(SEG$(BIN$,(LEN(BIN$)+1-I),
1)="1")):: NEXT I :: RETURN

200 A=A/2 :: BIN$=STR$(-(A-INT(A)<>0))&&BIN$ :: A=INT(A):: IF A THEN 200

210 BIN$=SEG$(RPT$("00"),8),1,1,16-LEN(BIN$))&&BIN$ :: RETURN

220 A=DEC+65536*(DEC>32767)

230 HEX$=SEG$(H$,(INT(A/4096)AND 15)+1,1)&&SEG$(H$,(INT(A/256)AND
15)+1,1)&&SEG$(H$,(INT(A/16)AND 15)+1,1)&&SEG$(H$,(A AND 15)+1,1):: RETURN
```

The same idea could be ported to FORTH but it would be much easier to
build a number base converter. Both TI Forth and Wycove already have
decimal and hexadecimal number bases built into the dictionary. We could
even add OCTAL (Base 8) as well. Here are the words we need:

: BINARY 2 BASE ! ;

: OCTAL 8 BASE ! ;

A message loop or two, a couple of screen prompts, an EMIT here and
there and we have a FORTH program that will print a number equivalent in
any of four different number bases. Wait a minute, mate! I'm not going to
write the program for you but this should stimulate an idea or two!
It's time for the annual cautions and warnings that cooler weather
brings with regards to computing. Make sure that your computer environment
is static-free. As the weather cools, the incidence of static electricity

The next couple of months are going to be a not-so-random byte column. Ever since I got the memory chips for the Quest that you so graciously gave to me I have been altering various programs to install into it. While I was visiting at Al Lawrence's, he mentioned a version of SPELLCHECK that he had that had been altered for Quest. Ever absent-minded, I neglected to get a copy before I left. That meant that one of the first tasks that I would have would be to alter my copy to work on the Quest.

Actually, the process wasn't as difficult as I had imagined. I had already modified it to run as a TI-Writer side option from F'WEB. One of the first alterations that I made was the one suggested in the F'WEB docs. Using DISK UTILITIES, a string search was initiated for the words 0460 0070. As per the docs, it was changed to 0420 0000. That allowed for an exit to the TI title screen instead of a crash when the program ended. With AUTO on, an easy re/entry to Quest was now possible.

I had previously installed SPELLCHECK in F'WEB simply by renaming its UTIL1 file to the two letter file designation in the slot in which I wanted it to reside. In my case, I renamed it DU. No problems so far! Now for the more difficult (?) part. I have my Quest divided into two drives ---DSK4 and DSK5. DSK4 (1536 sectors) contains all of my F'WEB options that I use frequently and is almost full. In fact, it is so full that the SPELLCHECK dictionaries will not fit. DSK5 has about 512 sectors free and it became the logical choice for the dictionaries. That is the background. Now for the particulars.

Luckily, the SPELLCHECK program is an easy one to alter. The second file load (ie. UTIL2) is hard-coded as is the dictionary load. Using DISK UTILITIES, do an ASCII string search in the first file of the series DU (formerly UTIL1) for the characters DSK. You will see "DSK1.UTIL2". Simply change it to the appropriate drive number ---in my case it became DSK4.UTIL2. Now, both of the files will load from the first half of my partitioned Quest. The second alteration (for the dictionaries) is just as easy. Just do an ASCII string search in the second file ---UTIL2--- again for the string DSK. You will see "DSK1.DICT1". All you have to do is change it to the appropriate drive number. In my case, it became DSK5.DICT1. The program automatically loads the second portion of the dictionary from the same disk drive so your task is completed once you have saved your altered files back to disk.

The biggest advantage in having SPELLCHECK on RAMdisk is the speed of the program. It now loads much faster and the execution and checking of a text file is also much faster than the original. I use the second part of my Quest as a cache for the temporary storage of text files, BBS downloads, etc. The result is that I copy the dictionary files into it at the start of a session (like this column) and then they are available for use whenever I need them. I might keep them there for a week or so until some other overriding need forces me to delete them to make room for something else. I have found it to be an excellent alternative to waiting for the physical drives to churn away while SPELLCHECK goes through another document.

Well, its getting toward the end of a page and I try to keep things as uncomplicated as possible for Joe. I'll have to think a bit about what to write for next month's column --- undoubtedly something to do with Quest or maybe some A/L since Ron K was nice enough to persuade me to try my hand at programming in it. At least my efforts will give he and Tony a few laughs but they had to start somewhere once as well.

Suggestions for this column are still (and forever) being accepted. Buffer full . . . 'Til next month.

FAR OUT (In the bush)
By Dick Schaydel

Good grief, said Charlie Brown, that sure was a bit of a shower we had (handy stuff sometimes), thought I had it beat though, I got a couple of two gallon buckets and scooping water out of the Culgoa, I made a mad dash overland to the Darling river. only that was bloody full of water too!, might have worked!. I was luckier than most around here abouts, I live on a bit of high ground, and although all around looked like a brown Pacific ocean, I didnt get much damage, lost a bit of cultivation but there wasnt much fodder on the rest of the ground to loose anyway.

Now that the water has receded a bit I can splodge my way to Lightning Ridge after a supply of fuel and tucker, then the environs of the L/R bowling club have to be explored again. I met a bloke there when I was getting the 'roo fencing who reckoned that I should have a go at making my own beer. It sounds like a good idea, it sure would scare Tooheys and the rest of that mob!

I think I'll go on holiday in a week or so. I've a mate who will come round and check on the automatic feeders and take charge of the place for a fortnight in exchange for a couple pigs. I think I'll take advantage of the situation and visit one of my mates in the Northern Territory. He has a place near Timber Creek and runs a boat safari operation on the Victoria River. He takes loads of tourists up and down the river to look at the local wildlife. Ever since those "Crocodile" films came out, everybody wants to see a Croc! Gordy reckons that as long as there is money in it, he'll keep it up. He wanted me to chuck my pigs and join him up there. Too much regulation for me, mate! I never did like that touristy stuff.

I was fossickin' about in my pile of disks and I came up with a program that is unique as far as I can tell. She's real beaut, mate! The name of this program is SUPERTRACE and it is written by the bloke who writes these Tigercub things. When I was writing programs in XB, the most difficult thing to do was to use TRACE to track down a program logic error. It all seemed simple enough. Type in TRACE in the command mode and then RUN the program. The trouble is, those flamin' line numbers flash across the screen so fast that you are never really sure where the problem in the program can be found.

SUPERTRACE does better! It works on a program SAVEd in MERGE format. It doesn't matter whether the program is BASIC or XB, it will break up the multi-statement lines before executing them. The line numbers are displayed as the program executes. So far, nothing spectacular. Wrong! One of the options of this program is to output to a suitable printer. Let me tell you, seeing something in hard copy is a lot better than squinting at those flamin' numbers flashing on the screen. Everything about the program is simple. All you have to do is follow what's on the screen and you can have a written copy of the program logic as it executes. A good piece of work!!!

A program that I use a bit is one written years ago by Dave Romer and John Clulow called TI-SORT. It works on a TI-Writer text file and will sort each line as if it were a record. There are two sort routines built in to the program and it will sort on two fields in ascending order. Using it, you can create database files with TI-Writer. Mailing lists are the first application that comes to mind. Just enter last name, first name, address, etc. The only thing that you have to keep in mind is to start the various "fields" at the same column. Load your database file and select the "fields" to sort by by entering the column numbers at the appropriate prompt. The file will be sorted and you can save the sorted file back to disk. Another good bit of work!

Living with pigs is always an interesting experience. I keep that lot fed with a high protein diet but they still have an appetitie for just about anything that comes along, I have a big boar that I call MacTavish that loves to eat lollies. It doen't matter what kind they are as long as they are sweet.

They are ALSO especially fond of snakes. Every so often, a snake crawls into the pen and one of the boars will usually get it. I don't know how they do it without getting bitten. I've never actually seen how they catch the flamin' things but I've seen them eating snakes.

Best go check on things before retiring for the night.!

HERE, THERE or ANYWHERE
~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Tony McGovern
~~~~~~~~~~~~~~~~

The simplest possible form of assembly program code to
think about is code that works at a fixed address for each
instruction and data word.   Every part of the program knows
exactly where every other part is, at absolutely fixed
addresses.   This is known as "absolute" code and is all that
could be written on many microcomputers more primitive than the
TI-99/4a, or with TI's cassette based Line-by-Line assembler for
the Minimem module.   Now by the time an E/A program file is
prepared it ends up in this form, and usually must be loaded and
executed at just the address specified in the 3 word file
header.

What I intend to do in this article, or short series
depending on how the length goes, is to look at how the TI-99
system allows the programmer to write a single piece of code
that may be executed at various addresses.   Assemblers for some
newer micros such as the Motorola 68000 series in fact make it
very difficult to write purely absolute code (though the
ingenuity of programmers can never be underestimated).

So let's look at the possibilities in increasing order of
complexity.   In TI-99/$a systems, code is normally produced by
the Assembler in the form of Tagged Object code.   Though the
9900 may now itself be old history, the TI tagged object format
lives on, not being restricted to 9900 code but used to express
for example TMS320 series digital signal processor (DSP) code.
Consult your E/A manual for full details as I don't really want
to go into all that here.   In fact the TI-99 assembler is a
direct carry-over of a mini-computer assembler, and supports
more facilities than the E/A loaders will handle.

Suffice it to say that in tagged object code, each byte or
word (compressed format) of code, data, or address information
is preceded by a tag character which the loader reads as an
instruction on what to do with the word.   The tag characters are
nothing more than 1..9, A..., and it is the job of the loader to
keep track of what is a fixed data word or instruction opcode
(they are the same thing to a loader) and what is an address
reference that needs to be adjusted at load time.   In ordinary
or uncompressed object code the words are written out in
hexadecimal and the Dis/Fix 80 files can be read and edited with
a text editor.   Compressed object code keeps the tags as
alphabetic characters but expresses the word after each tag as 2
machine binary bytes.   The only practical way to read or edit
this is to use a sector editor on disk in its hexadecimal
display mode.   Tags "9" and "B" indicate absolute addresses and
data or instructions, and "A" and "C" indicate relocatable.

Absolute code is produced if an AORG directive is used in
the assembly source code.   You are instructing the loader to
place the code exactly as is, at the definite address specified.
The TI system however is biased towards the use of relocatable
code especially in conjunction with Basic or XBasic.   If no
specific directive is given the assembler produces tagged object
code that the linking loader recognises as relocatable.   The
loader then decides where it can put the code, and adjusts the
relocatable addresses and data accordingly as it puts them in
place.   Once it is loaded you cannot tell the difference between

this code and data in memory and the code and data in memory if
it had been placed there by a absolute load which happened to be
to the same address. Only the loader knows and it keeps track
only of relocatable programs so it can figure where to put the
next relocatable block.

All of the above is familiar stuff for any assembly
programmer. What we really want to look at is how to get code
to execute correctly at addresses other than those at which it
was initially loaded. Why on earth would you want to do that ?
One reason is that you may wish to have code which is placed on
one or repeated occasions to execute elsewhere, commonly in the
PAD for speed or to avoid memory bank-switching traumas as with
Myarc RAM-Disks.

There are two ways to generate such code. One is to
trick the assembler into generating code that runs correctly at
the address at which it is ultimately placed, rather than at the
address at which it is initially loaded. The assembler itself
provides the way with the DORG (for Dummy ORiGin) directive.
The more thorough-going way is to write code which genuinely
does not care which address it is at, known as position
independent code. This often turns out to be very easy for
short pieces of code and rather more tricky for typical longer
program segments.

First we will look at the DORG directive. When the
assembler encounters this it quits producing any code but still
keeps track of addresses following the DORG statement. When it
next encounters a AORG or RORG it goes back to producing code.
The usual examples you will find in TI source code are to do
with setting up data fields or workspace register names in areas
of memory not included in the bounds of the program. Simple
cases like these can also be handled as EQUates.

```
*   Address generated by DORG     |   *   Addresses by EQUates

        DORG >8328                 |
BUFMWS  DATA 0          Wsp & R0   |   BUFMWS  EQU >8328      Wsp & R0
TBLND   DATA 0          R1         |   TBLND   EQU >832A      R1
EDBFND  DATA EDITBF     R2         |   EDBFND  EQU >832E      R2
        DATA SQSBUF     R3         |   LINEN   EQU >6330      R4
LINEN   DATA 0          R4         |
        RORG                       |
```

The two segments of assembly source above both do the same
job of of defining addresses for a workspace to be set up in PAD
RAM. As far as the rest of the program is concerned both pieces
of code have exactly the same effect in setting up addresses for
part of a workspace for use by the program in PAD. At that
level of complexity there is not probably much to choose between
them, but the one on the left is better programming practice –
for the programmer – the assembler doesn't really care. The
code on the right causes 4 entries in the symbol table. So does
the code on the left, and the assembler has to do more work to
get that done. Why then is it better ? The reason is that it
shows more clearly what is intended for this workspace, and very
succintly and clearly reminds the programmer of initialization
code that will need to be executed. Remember that though there
are data statements in the DORGed code, no object code is
generated that would place data at those addresses. That code
has to be written separately in either case as part of the
program.

If the DORGed code were rather more complex it would become a real pain to to keep track of where the EQUates should go. It is much easier just to make a copy of the code using the editor. Let us look how to generate a block of code for the DORG directive from the actual code which has to be carried in at one address and later moved and executed at the address set up by the DORG directive. An example of that is in the Funnelweb Assembler, where the filename entry code has to be brought in latched onto the end of the program file AT (to save slowing the loading and to save wear and tear on disk drives if another program file were generated), but for various reasons needs to execute elsewhere in memory.

This is done by having one copy of the code DORGed at its actual execution address to establish the labels, and a second modified copy for which the assembler generates code in the main block. One way or another the program shifts the code as a block to its execution address. Some care however has to be taken with labels. Addressing of a memory location is done in three ways in the 9900 processor. The shortest is register addressing, R0 through R15, which at 4 bits is short enough to have two of them in one instruction. The processor goes to memory at a full word address obtained by adding this offset to the Workspace Pointer. This does not really care about where the code is executing from as long as there is no conflict. The 9900 family is different from most other micros in this regard, though the approach is re-appearing in various forms in some RISC processors. The next shortest is Program Counter (PC) relative addressing as found in jump instructions or for data words in Immediate instructions such as LI R0,>100. In the JMP type of instruction 8 bits are used as a word offset counter from the word following the JMP or other such instruction. This one is inherently dependent on the current PC value, and if the JMP is to an assembly label, this must be in the right place relative to the actual execution address. The last of course is just the normal full 16-bit address which takes up a whole word.

So we have a DORGed copy of the code which has the address labels (well.., almost all of them) but generates no actual code, and the copy which generates the code but needs only local PC relative target labels for Jump instructions. It is a good idea to use dummy labels for other addresses in the code for clarity. The code example is for illustration only.

```
*  Copy for address generation  !  *  Copy for code generation
*                               !  *
        DORG DADDR              !
ENTCOD  EQU  $                  !  CODSEG  EQU  $
        MOV  R11,R10            !          MOV  R11,R10
        BL   @SUBR1             !          BL   @SUBR1
        LI   R0,>20             !          LI   R0,>20
ENT10   MOV  *R3+,@VDPWD        !  ENT10X  MOV  *R3+,@VDPWD
        DEC  R0                 !          DEC  R0
        JGT  ENT10              !          JGT  ENT10X

        B    *R10               !          B    *R10

SUBR1   EQU  $                  !  SUBR1X  EQU  $
        ..                      !          ..
        ..                      !          ..
        RT                      !          RT

        RORG ( or AORG )        !  CODLEN  EQU  $-CODSEG

*  Program continues            !
```

The example may be short and artificial, but it does show all the essential features. The BL instruction uses a normal full word address as DORG generates the address for the SUBR1 label at its correct ultimate location in the DORG block. In the regular segment the BL must also be to SUBR1 and the SUBR1X label is for information only and in fact may be omitted entirely, at the cost of clarity in the code (while you could BL to anywhere in the DORGed BL as long as the instruction length remains the same). The next label in the DORGed code is a PC relative jump target. For clarity we put it in the right place, but it doesn't really matter (and even a JGT $ could be used). Over in the actual code the ENT1OX label is the one that really sets up the correct jump on JGT. If you forgot to change the name of the label in the JGT instruction, the assembler will generate an Out of Range error. It is better to use a new label for the local Jxx rather than to count out the offset and use JGT $-6 as the presence of the label makes it easier to dompare the DORGed code with the RORG or AORGed version. If you have an 80 column system (AVPC or Mechatronics) then the split-screen dual file View function of DiskReview is very handy in tracking down possible problems.

If you had gotten carelessly overenthusiastic and used SUBR1X as the label for the BL in the generated code, then the assembler would not flag an error. This can be an insidious problem in code testing, since the original copy of the code might still be in place early in program operation and only corrupted later by data or whatever. Finally VDPWD is a full word address independent of the DORGed block. As a last reminder if the code is changed, matching changes must be made to both copies.

Next installment will be about writing strictly position independent code for the 99/4a and on some precautions for writing subroutines so that they may be called from any workspace.

Funnelweb Farm
May 30th / 1990

Who's the Redhead!

Well, here we are - next month already.   IF anyone bothered with the humdrum happenings at the barby and/or the restaurant AND tried to work out the answers AND couldn't, here they do be.   Well, not exactly.   If anyone tackled the Who's the Redhead? they must surely have found the answer as it was too easy, so I won't add insult to injury (self-inflicted) by providing the answer.   As for the teetotaller - this one is not as straightforward as it first seems.   Of course I have the advantage of word-of-mouth from the IWP, and decided to slip in a spot of deception.   However, those who permuted the data and came up with Libby can be excused for not knowing that Libby is not a teetotaller but a real little rager who had just happened to draw the short straw for the driver position that night. 'sides which she has a Tarago.

DRAWING UP A TABLE IN TI-WRITER.
by GARY JONES HV 99ERS.

The following is a small transliteration file which allow you to draw up a table or any type of regular boxes in text while using "TI-WRITER".

These char values shown below are for an EPSON compatible type printer.

TABLE AS PRINTED.

TRANSLIT FILE	USE	IS
.TL 33:184	!	r
.TL 60:181	<	—
.TL 62:185	>	⌐
.TL 96:182	'		
.TL 91:179	[	⊢
.TL 92:177	\	⊤
.TL 93:178	]	⊣
.TL 95:186	_	∟
.TL 123:176	{	⊥
.TL 125:187	}	⌐
.TL 43:175	+	+
.TL 34:27,49	"	7/72
.TL 37:27,50	%	1/6

TABLE AS IN TEXT.

```
!<<<<<<<<<<<<<\<<<<<\<<<<>
!TRANSLIT FILE! USE ! IS !
[<<<<<<<<<<<<+<<<<<+<<<<]
!.TL 33:184    !  !  ! ! !
!.TL 60:181    ! < ! < !
!.TL 62:185    ! > ! > !
!.TL 96:182    ! ' ! ' !
!.TL 91:179    ! [ ! [ !
!.TL 92:177    ! \ ! \ !
!.TL 93:178    ! ] ! ] !
!.TL 95:186    ! _ ! _ !
!.TL 123:176   ! { ! { !
!.TL 125:187   ! } ! } !
!.TL 43:175    ! + ! + !
!.TL 34:27,49  ! " !7/72!
!.TL 37:27,50  ! % !1/6 !
_<<<<<<<<<<<<<<<<<<<<<<<>
```

The above char may be altered if any are required for text.

As normal chars they may-be also altersd by the other control chars with-in the printer!-
            e.g.(i) ENLARGED ⌐
                (ii) EMPHASISED ⊥

NOTE. Use " to close up the line spacing to allow for continual vertical lines,
 Then % to open up for normal line spacing.

HINT. Set up the TAB function to tab to the next vert line in the table.

TO SET UP THE TRANSLIT FILE.

The file is set up as a normal text file!-

(1) Enter the editor.
(2) Type each transliteration line followed by a <carrage return>.
(3) Save file under a FILENAME.
(4) Don't forget to .IF DSKn.FILENAME when using the file in a text.
(5) To reset the chars for normal use, set up another tranliteration file.

        e.g. .TL 33:33
          *   .TL 60:60
          *   .TL 62:62
            etc,etc

        this is to be included after the table in the text and the chars are required as standard chars.

I hope this small note will help when drawing up tables.

14

NIBBLEBYTES
Al Lawrence

7 Years on and fame at last for the TI99/4A, Tony and Will McGovern. In a 10th birthday supplement supplied with the May'90 Australian Personal Computer magazine. There are chapters on the 10 Most,Notable,Best,Quotable etc. and in a segment titled "The TEN Years of Frenzy " John Sterlicchi chronicles notable events in the PC Industry. A mention that Time magazine featured the PC as Machine of the Year for 1983, instsead of the usual Man for the Year, and the bleak day in October when Texas Instruments left the home computer market,dumping the TI99/4 and also the bad strategy in their marketing the TI, comes the following extract :-

" More than six years later, the computer is neither gone nor forgotten. There are user groups all around the world and a year or so ago I discovered that two Australians were doing their bit to keep users happy and productive, University of Newcastle physics professor Tony McGovern and his 16 year old son Will created Funnel Web a word processing package named after the infamous spider. At the time,one enthusiastic US user told me that the latest version, FunnelWeb 4.0, was the most single important programming effort for the TI. Another said " :-

"Funnel Web is the king of the TI word processors."

Well done to Tony and Will, Now it is even better with the Version 4.30. which is now released and in the HV99'ers Library.

More developments in the 80 col. Disk Review. Now a full management and sector editor system,takes Funnelweb closer to full blown DOS status without having to leave the Web. Using it as the BOOT disk, you can opt for Disk Review, F'Web, or User's choice on firing up the system even if you don't have a RAMdisk and the J.Johnson Miami MENU. Having an 80 col. card access is one more reason to stay with IT as Tony and Will push us beyond PARSEC.

Hurry up Peter Gleed with my AVPC card,it will be worth the long wait.

Tony has also updated the 40 col F'Web version, so it is truly worth owning and using a TI 99/4A for a few more years. And as they say there is more hardware and software being developed for it since 1983 than may have been possiable if TI had stayed on and kept their closed Architecture and Copyright policy.

Good news for those with a HardDisk, Tony and Will now have one and where last seen climbing the tree of knowledge towards the interior apex of a simulated black hole.

Make sure you buy or read a copy of June MICROpendium as Bob Carmany is doing a review of the 512k RD200 QUEST RAMdisk developed by Neil Quigg who was the HV99'ers most inovative hardware developer of '89.

Ron K.is still developing software for the above and I hope he gets the credit he deserves for bringing us closer to the edge of no return before we disappear for ever into the black abyss.

TIsHUG have a good buy of pre Loved WANG monitors for around $130 and a full 3 month warranty. They are taking orders NOW, see Tony if you are interested.

```
      ~~~~~~~~~~~~~~~~~~~
      W-AGE/99 * NEW-AGE/
      99 * NEW-AGE/99 * N
      EW-AGE/99 * NEW-AGE
      /99 * NEW-AGE/99 *
      ~~~~~~~~~~~~~~~~~~~
```

* by JACK SUGHRUE, Box 459, East Douglas, MA 01516 *
                    #5

Anyone in the TI World owning a disk system at least a month and has
not yet contacted Jim Peterson at TIGERCUB Software is certainly leading
a severely deprived life.
  Jim has the largest collection of stuff at the cheapest prices
possible for our amazing 4A.   He couples this "best for the least"
business with a fantastic knowledge of the machine and a kind, generous
spirit.   No one knows the BASIC and XB workings of the TI better than
Jim. He is an expert in everything!
  Mr. TI, as he is known by his thousands of admirers, seems to take to
his computer the way Jean Henri Fabre took to ants. He is meticulous
and creative and understands the soul of the 4A. I'm still in awe of
his skills and dedication and influence.
  There is no one in the entire TI World (unless he or she has been
hiding under a rock since the 70s and has just been handed a machine)
who has not felt the influence of this mild-mannered, modest man.
  I've come across hundreds of programs with his fingerprints on them:
programming enhancements and tools he has given to us.   For years Jim
gave newsletter editors free tutorials called "TIPS FROM THE TIGERCUB"
which were (and are) so jampacked with wonderful programming goodies
that it is hard to imagine what TI life would have been without this
marvelous source.   He still sends these "CARE" disks to sharing
newsletter editors.   He has also written the ultimate tutorial on
programming in the form of subprograms that can be easily merged into
any XB program (including a subprogram that makes BASIC programs into XB
programs). These loaded disks of subprograms (called NUTS & BOLTS) can
be purchased from him and readily used with your own or other programs.
These Tigercub touches are what I see on almost every good XB program
written for the TI in the past half decade."
  Jim has also written so many programs for the Public Domain that we
just take for granted that these kinds of programs have "always been
there."   They weren't.   Until Jim gave them to us. There isn't a user
group library in the world that doesn't have heaps of programs from
Mr. TI.
  He also writes numerous "commercial" pieces of software. "Commercial"
only in the sense that they are for sale. They were low-priced and of
high quality in the heydays of the 4A when everything was high-priced
and too often of extremely poor quality. Years later, I still use a
large number of his programs in school (SYNONYMY, MECHANICAL APTITUDE
TEST, SCRAMBLE, SQUINCH (a fiendish word game), to name a very few).   I
wouldn't like to be without the other Tigercub utility and game programs
I've enjoyed so much over the years (particularly the unendingly
fascinating NUTS & BOLTS disks, which I had the honor of demo-ing at a
recent computer fair. [I had the greater honor of meeting Jim at the
great TI fair in Lima, Ohio.] The man's remarkable and is universally
liked (which is remarkable unto itself).
  [Jim's three NUTS & BOLTS disks (with a descriptively succinct
manual/tutorial) are now only $10 each. His five disks full of "TIPS
FROM THE TIGERCUB", a newsletter editor's Godsend, are only $5 each.

ı b
```

His 120 original programs (a refundable $1 for the catalog) are now just $1 each!]

Praising Jim's efforts on our behalf is not the purpose of this article. (It's just impossible to write about Tigercub without doing so.) The purpose of this article is to tell you about the latest goodies to come out of Tigercub. Jim, because of his huge number of TI contacts (without a doubt more than anyone else in the world), has been able to put together the largest collection of Public Domain and Fairware programs, files, and templates in existence.

This PD extavaganza can be dipped into by sending a refundable dollar for this catalog. ($2 for both the Tigercub and TI-PD catalogs.) Within is an unbelievable world of goodies. A 4A maniac's paradise! At only $1.50 per disk! Not per program. Per FULL disk! And that's postpaid!

These disks do not contain a pile of junk you'll never use, either. They are selected from the thousands Jim has in his library. And they are catalogued and sub-catalogued and regrouped.

An example: Interested in music? Those are the 600 series. What kind of music? Well, remember those great graphic/music combos of Sam Moore? 600 is a disk called "SAM MOORE MUSIC #1" (341 sectors). It has 11 super selections on it. But there is also a "SAM MOORE MUSIC #2" (343) and a 3 (348) and a 4 (337) before #604 moves to "BILL KNECHT HYMNS" (334) and so on.

You get the picture. You get a disk full of the kind of things you want and can use: educational programs, graphics, printer utilities, typing, health, you name it. Games are broken down into so many categories it's amazing. Three disks of just CARD games! All programs now run in XB and all come with Tigercub's famous Loader, forerunner of all the good loader programs found elsewhere. Jim has games broken down by specific type: "ROAD CROSSING GAMES", KEYBOARD MANEUVERING GAMES", "Q*BERT GAMES", "FORMER COMMERCIAL GAMES", "EASY GAMES FOR KIDS", "KING KONG TYPE GAMES", "TWO-PLAYER JOYSTICK GAMES" (there are loads of one-player), "GERMAN GAMES", on an on, page after page.

There's even such esoteric stuff as "LIGHT PEN PROGRAMS (including a disk file which teaches you how to make your own light pen).

There are disks of programs about Chemistry; Hi-res Drawing; Physics; Children's Programming with Speech; Sorts, Scrambles & Searches; Auto-loaders; Calculators & Converters; Astronomy, Religious Programs. The list seems endless.

The catalog gives you the full listings of the files on the disk: "FINANCIAL PROGRAMS" (356 sectors) includes the following selections with authors where known: Amortization Schedule (M Holgers); and #2 (J Roche); Compound Interest (C Good); Estate Tax Securities (R Shumaker); Debt Calculator (K Romstedt); Financial Math (C Ehninger); Financial Statemnt Ratio Analysis (C Colton); Investment Analysis (A Robertson) AND 15 more!

Just the work and time involved in the collecting , reviewing, selecting, debugging, sorting, creating full disks, cataloguing, printing, and distributing must be incredible. To charge $1.50 a disk is the greatest TI giveaway of all times.

Order the catalogs today; then, after you wipe the drool off the table, order as much as you can to show Mr. TI how much you support his endeavors. TIGERCUB Software, 156 Collingwood Ave., Columbus, OH 43213.

[If you use NEW-AGE/99 please put me on your exchange list.]

# Thank You !! Bob !

Amongst the "pressies" that Joe Wright gave me was a book of facts and figures about Newcastle and its environs. It was very interesting reading. For example, the highest temperature recorded in the Newcastle area was 41.3 Celsius and the lowest was -.5 Celsius. These statistics might interest someone but they sure don't do much for me!! There were some facts and figures that were just briefly mentioned (like wine production) and others that were left out entirely (probably an oversight). However, having an eye for observation, here are a few that I observed from my stay in the Hunter Valley and Queensland.

1) With few exceptions, Hunter Valley blokes have a penchant for growing facial hair. In fact, I would estimate that more than 50% of the UG have some sort of adornment. There are moustaches and small beards (Alby). There are short-cropped beards (Peter and Tony), there are semi-full beards (Richard Terry), and there are full-blown reforestation" effort" (Woodsy). Those who don't currently have beards or moustaches used to (Al Lawrence) or want to grow one (Joe Wright?).

2) Hunter Valley leads Oz in the per capita consumption of Port. This is mainly due to the Herculean efforts of Al Lawrence, Joe Wright, Richard Terry, and Brian Woods. I tried to help while I was there but there is no keeping up with that lot! The very-civilized practice of a cuppa and a glass of Port is one I have adopted here. I am still having trouble adjusting to it after morning Tea, though.

3) After a two day sampling, supervised by Brian and Al Lawrence, I determined that the NSW drop --Tooheys Draught -- is superior to all others in Oz. The margin, however, is very slim. The Queensland entry -- Powers -- is a very, very close second. Remember, this was based on lots and lots of experience -- and lots Victoria Bitter, Swan Lager, Fosters, etc., etc.

4) Oz tucker is fair dinkum five star cuisine. From the "sausage sizzle" at Joe's place to the Ruth Cavanaugh's scones and jam, everything was really first-rate. Of course, there was the "odd kilo" of prawns thrown in from time to time. Even the Vegemite and toast at Woodsy's was palatable. It takes much more effort to button my pants and fasten my belt since I got back from Oz!! I could get used to it, though.

5) I have also determined that 100% of Novocastrians are friendly. How's that for a statistic! Everybody says "G'day" and we all ended up being mates before the trip was over.

6) Even the accent isn't too bad. The only problems that I had was trying to understand the conversation between Tony and Ron K and that was because they were "speaking in tongues" ---Hex!!

To sum the whole thing up, I would like to quote from a postcard that I saw --- "The Beer is great and the natives are friendly!"