



APPLE - BUG



The Newsletter of **APPLE-Q**: the Brisbane Users' Group
Post Office Box 721 - South Brisbane - Qld. 4101

[SEPTEMBER 1987]

[In this issue]

Editorial	< >	Vice President's Notes
Secretary's Notes	< - >	BBS Notes
Software Library	< - >	Page Right
Feature Article	< - >	Programming Languages
Advertisements	< - >	Trading Table
Nomination Forms	< - >	Membership Renewal Form

NEXT MEETING: - 20th September 1987

<<< CREDITS >>>

Tony Truscott	< >	Away on Holiday
Graham Black	< - >	He typed it all in
Vince Crosdale	< - >	He typed some too
The Members	< - >	For their contributions
ZARDAX II	< - >	Word Processing
EPSON LQ-1500	< - >	Typesetting
The Hooper Centre	< - >	Printing & Distribution
The APPLE Computer	< - >	The reason for it all

[Executive Committee]

Terry Sweetser	President	Ph.(07) 203 4210
Bernie Benson	Vice-President	Ph.(07) 345-1545
Vince Crosdale	Secretary	Ph.(07) 351 3090
Eric Conolly	Treasurer	Ph.(07) 261 1860
Graham Black	Registrar	Ph.(07) 883 1525
Tony Truscott	Editor	Ph.(07) 266 4795
Peter Lane Collett	Assistant Editor	Ph.(07) 398 9282
Richard Eggesfield	Hardcopy Librarian	Ph.(07) 349-8888
Bob Godbehere	Software Librarian	Ph.(07) 808 3892

[Bulletin Board System (BBS)]

Apple-Q BBS : online 24 hrs
 Telephone : (07) 284-6145 (DATA)
 : (07) 883 1525 (VOICE)
 Baud Rates : 300, 1200/75, 1200 and 2400 (CCITT and BELL)
 Data Specs : 8 Data bits 1 Stop bit No Parity (Full Duplex)
 Sysops : Graham Black & Vince Crosdale

BBS equipment : Apple //e 128k
 : 1 x 5meg Profile Hard Disk (it finally arrived)
 : 1 x 800k 3.5" drive
 : 1 x 1meg Ram Card
 : NetComm 1234A Smartmodem
 : Epson LX-80 Printer

Calls to the system : 6049
 Registered Users : 293 (as of noon 3rd September)

SYSOP stands for: SYStem OPerator
 BBS stands for: Bulletin Board System

[What's When]

at the Hooper Education Centre Kuran Street Wavell Heights

Sunday 20 September 1987	Monday 21st September 1987
Open Day	Committee Meeting
Hours: 9.00.am. till 4.30.pm.	Starts: 6.30.pm.

Sunday 18th October 1987	Monday 19th October 1987
AGM - Birthday Party - Open Day	Committee Meeting
Hours: 9.00.am. till 4.30.pm.	Starts: 6.30.pm.

[Membership Fees]

Joining Fee: \$10 Adults/Family: \$20
 Students/Pensioners: \$12 Corporate Membership: \$50
 Associate Membership: \$ 5 plus \$5 Joining Fee. (Pro rata till MAY 88)
 (Students under 21 years - Pensioners on production of Pensioner Card)
 (at the discretion of the Executive Committee)

[Editorial]

The Annual General Meeting and Apple-Q's birthday are only one month away. In keeping with our 'traditions' we will have our Apple ice cream cake, apple pie and cream and Apple Cider (*non-alcoholic of course*). Very little serious computing is to be done on the day and wives, children and friends are all invited to share the festivities. The formal portion - election of committee is timed for around 12:30 immediately followed by ice cream cake and cider. To round out the day it is hoped there will be no shortage of volunteers to clean up the area so that the Hooper Centre will get a good impression of the new committee.

If you are interested in becoming involved with the running of the club, then please fill out one of the Nomination Forms in the back of this newsletter, indicating which position you would like to hold. The committee would like to see more members take an active part in the running of the club. There are two Nomination Forms, and others will be available at the Hooper Centre on Open Day.

Please take note:

Members are required to display their membership cards on their equipment so as to allow identification of visitors using club facilities and causing inconvenience to bona-fide club members. Also, all club members and visitors should try to make the effort to sign the book placed near the door to the Hooper Centre.

Once again I find it necessary to remind the members that the rooms are being left in a frightful mess each month. The coffee machine seems to be the major target for people bent on this course of action. The Hooper Centre allows us to use the coffee machine for a minimal charge, and we have to pay for the ingredients that are used or wasted each month. This includes the plastic cups, coffee and sugar. Unfortunately, the major offenders in this case are the numerous children brought along to the meetings by parents and left unattended to find their own methods of amusement. Therefore, we would make a request to all parents to please ask their children to steer clear of the coffee machine, and to all members using the machine to be more conscious of the wastage factor. We are not singling out one group in particular, instead this message is made necessary by a number of complaints about the amount of materials being wasted.

[Vice-President's Notes]

by Bernie Benson

We have reached the end of the club year, and it is this time of the year that you as a member of the club should be considering whether you are prepared to give some of your time for the benefit of the club. Most of the positions involve some extra effort on your part. However, the old excuse that you cannot spare the time doesn't stand up because if you are fairdinkum enough you will find the time.

The present committee does not have any exclusive hold on their positions and all jobs are up for nominations.

Think about what you can do or would like to do so as when nominations are called for the coming year you will be prepared to accept a nomination.

If you are not prepared to accept a position on the committee at least cast your vote for those members who are nominated.

[Secretary's Notes]

by Vince Crosdale

Well, after last month's mighty effort by yours truly, it feels great to back to familiar ground and to have something to tell you good people. Last month's lack of correspondance seems to have been something of a freak, if I compare it with this month. We have recieved numerous letters and calls from places distant to ours, including Perth and Tasmania. It just goes to show how far we spread our wings.

Mail this month has consisted of a number of letters from people making enquiries about the club and the usual magazines. For those of you who are not as yet aware, we are now recieving the newsletter from the Tasmanian Apple Users Group in Hobart. Theirs is an excellent magazine, and contains a lot of little tips that we have not seen up here as yet. I recommend anyone using our library to have a look at this worthy addition to our ever growing collection.

Also, thanks must go out to Jim Fraser, Secretary of TAUC. Jim's son decided to build the auto-fire circuit which was contained in our newsletter last month, and discovered that there appeared to be some sort of problem with the circuit as displayed. Not to be beaten by this, Jim then rang Graham who placed him in touch with myself. I had only the circuit diagram I was supplied with to go on, and eventually we decided it would be easier to ring the author of the article. So then Jim rang the author and found the solution to his problem. We hope to get hold of this solution and print the errata for you in the next edition. Thanks Jim, for making us aware of that one, but I'm glad I don't pay your phone bill!

Another matter of interest that has come to light. As most of you will be aware, we have placed a minimal charge on the copying of the club software. There is more about this elsewhere in the newsletter, but I will add this. The proceeds from this endeavour have been placed to good use for you, the members. We have taken out a subscription to the International Apple Core (IAC). Included in this deal, is a monthly disk of software for the Public Domain Bank, and also a copy of NIBBLE magazine for the library. The money taken in from this copying, plus a proportion of the club funds will now be spent upon magazines and software for the Club. The software library has fallen rather out of date, and we are now attempting to rectify this situation. On the matter of the Public Domain bank, thanks must go out to Sheryl Mann for having kept copies of a few of the disks in the bank. The originals became corrupt for some unknown reason, and luckily Sheryl had copies of the bad disks that we were able to copy back. Thanks Sheryl.

There is just one other matter. I hate to end on a sour note, but unfortunately, it has become necessary. After the meeting last month, a quick tally of the figures at the Trading Table was made, and it suddenly became apparent that we were short a packet of disks. Now, after much recounting of numbers and retallying of monies, we are still in the same boat. We cannot accuse anyone of this theft nor can we prove that a theft was actually committed, but we do know that we are a box of disks down. The Trading Table is run for the advantage of the members, and if items are stolen from the Table then it is to the detriment of all concerned. Now I would like to think that this sort of thing does not happen at our Open Days, but I may have to rethink my stance on this one. So we would just like to tell all members, that if this sort of thing happens again, then we will be forced to have a long think about the Trading Table and the displaying of items.

Well, that's about it for this month. Please remeber that contributions for the newsletter are always gratefully recieved and thanks to all members who have taken time to contribute articles to us. All articles for the newsletter may be forwarded to us either on disk or typed. Articles may be

handed to us at the Open Day, and if you submit on disk, then all care will be taken and your disk returned.

So until next month, Good computing.

[**BBS Notes:**]

by **Graham Black**

A new file transfer section has been purchased for the Club's BBS following last months Committee meeting and the resolutions passed are outlined below. This was due to the necessity of a credit type system for the uploading and downloading of files onto the BBS. With this new set-up, it has become exceedingly easy for the Sysops of the system to perform a re-configuration of the file transfer area, and incorporate into it a section allowing use of the Club's vast bank of Public Domain Software. With the previous segment, it was difficult to allow for the continual changing of the files contained in the Public Domain areas, but this problem has now been overcome. We will be placing a new volume on-line each week. It has taken a great deal of time and effort to achieve this, and as the new volumes are prepared, they will take their turn on the system. All members should note that if they intend to download files from the Public Domain areas on the system, then we recommend that you first take the three files contained in the area designated 'Associated Public Domain Files'. These files are necessary to allow for the renaming of the files contained in the four Public Domain areas, as all of the files operate under DOS 3.3 and the BBS operates under ProDOS. If anyone is having any trouble with the downloading and renaming of the Public Domain software, then please get in contact with Vince at the next meeting or at home during reasonable hours.

A Resolution was passed at the August Executive Committee Meeting. Here are the results of that resolution, as posted on the BBS, and viewed by everyone when they logon for the first time:

New User Information

USER ACCESS to the BBS will be \$5 per Annum, with a \$5 joining fee. The Financial Year for the BBS will be from June, to May of the following year. The Fee will be PRO-RATA. There are three categories of User to this system. These are as follows:

NON-PAYING USER : You will be allowed access to the Public Boards only
 Limited File Transfer (on the credit system)
 30 minute time limit

ASSOCIATE MEMBER: (\$5 joining Fee & \$5 per Annum)
 Access to the Public Boards
 Games
 File Transfer (the General section only)
 50 minute time limit

APPLE-Q MEMBER : Access to the Public & Members Boards
 GAMES
 Full File Transfer
 60 minute time limit

APPLE-Q MEMBERSHIP FEES:

- : \$10 Joining Fee and
- : \$20 Adults
- : \$12 Student/Pensioners per annum
- : \$50 Corporate Membership
- (at the discretion of the Executive Committee)

Please note that only Financial Members of Apple-Q will have FULL ACCESS to this system. Apple-Q members are reminded that if they are found to be UNFINANCIAL, their MEMBER ACCESS will be reduced, and they will only be able to access the Public Boards, with no access to the games.

[Software Library]

By resolution of the Executive Committee, there is now a copying fee for the Public Domain Software. The fees are being used to raise funds to purchase more software for the Library, and some disk drives for the copying of these disks. At the moment, the Software Librarian's drives are being used, and worn-out. The fees are as follows:

50 cents per side, with a minimum charge of \$2.00 (4 sides)

I.A.C. Volumes: (15-20)

The files are divided into sections, and the files in each section are required for the particular applications. The Prefix before each file name represents the file type. i.e. B=Binary T=Text A=Applesoft I=Integer. Each section is separated by a string of -'s and each volume is separated by a string of ='s.

Volume # 015 - Disk 1 : FREE SECTORS = 9 (2.25K)

A TULSA MENU	B MENU.1
B MENU.2	A ARITHMETIC TAC DOUGH
I AWARI	I DECISION
B COMPUTERS.DEC SN	A HERB'S CALCULATOR
B HERB'S CALCULATOR DISPLAY	A INVERSE FIELD INPUT
B INV.INPUT.INST	T LIFE EXEC
I LIFE (LOMEM-3072 HIMEM-8192)	I MADLIB
I OTHELLO	A SCREEN PRINT INSTRUCTIONS
B SCR.PRINT.INST	B SCREEN PRINT
A SCREEN WRITE INSTRUCTIONS	B SCR.WRI.INST.1
B SCR.WRI.INST.2	B SCR.WRI.INST.3
B SCREEN WRITE 2.1	A SCREEN EDIT
I STATE CAPITALS	A TEXT EDITOR INSTRUCTIONS
B TEXT.ED.INST	A TEXT EDITOR.50
T TEXT EDITOR TESTFILE	I TOWERS OF HANOI SOLVED

Volume # 016 - Disk 1 : FREE SECTORS = 148 (37K)

A THE GREEN APPLES	A MEMORY GAME
A PRODUCTION OF LIGHT	A SPELLING TEST
T SPELLING LIST	A TOM'S TRASH COMPACTOR
A RUTHER	A TOM'S MATH DRILL

A TWO DRIVE COPY	B RWDISC
B DOS POINTERS	A DAN'S SWITCH
A DAN'S OPTION WRITER	A ALGEBRA 1
A SIMON	A GAMBLER
A SHIP CAPTAIN CREW	B H-DICE
A STATIC ELECTRICITY	A APPLEWRITER READER
B TEXT.APNOTE-B2	A TOM'S DISK MODIFIER

Volume # 017 -- Disk 1 : FREE SECTORS = 89 (22.25K)

A HELLO	I LIFE
I DATA DISPLAY AND SAVE	I NJ LOTTERY
I STATES AND CAPITALS	I BINGO
I AUTOMATIC BINGO	I BINGO CARD
I TITLE DEMO	I TWINE (TYPE GR FIRST)
I BEGINNING	I COMPUTER ART
I WINGS(TYPE GR FIRST)	I TWO CIRCLES
I PHILA. PINBALL	I FAST MEM TEST
I SLOW MEM TEST	I PHILA. COLOR ORGAN
I CRYPTOGRAM	I MUSIC
I INSTRUCTIONS	I SLOTS
I RANDOM LINES	I THE HART PIANO
A HANGMAN	I BLACKJACK
I STARWARS RESCUE	I HANGMAN 2
I SKUNK	I MOON LANDER
I LAS VEGAS SLOT MACHINE	I NEVADA CRAPS
I COLOR MATH	I BATTLE OF NUMBERS

Volume # 018 - Disk 1 : FREE SECTORS = 187 (46.75K)

THE PROGRAMS ON THIS DISK ARE FROM W.V.A.C.C. 535 HIGHLAND DR WENATCHEE WASH. DAVID HOFFMAN

A HELLO	B PIC. #1 (HI-RES PG.2)
A SPACE MUSIC+ INTRO	A UP & DOWN THE SCALE
A JOYSTICK/PADDLE TESTER	A SOME WHERE OVER THE RAINBOW
A APPLESINE	A ASCII SCREEN DISPLAY
A TUNNEL VISION	A TWO 3-D SHAPES
A PLAY THAT TONE AGAIN SAM!	A GRAPHING FUNCTIONS IN HI-RES
A SNAKE ARCADE GAME	B SNAKE LOGO
A AMPERSAND LIST UTILITY	A AMPERSAND RUN UTILITY
A TWIRLING CURSOR	A LIST THIS FILE BEFORE IT'S RUN
A TRICK FILE NAME UTILITY	A AMPERSAND KILL-CATALOG UTILIT
A HI-RES SOLAR SYSTEM SIMULATOR	A FILE ARRANGER UTILITY
I WENATCHEE APPLESTAND (GAME)	A LOAN AMORTIZATION TABLE
A ^^255 HI-RES COLORS DEMO^^	A PAUSE (WORKING) ROUTINE
A BILLBOARD DISPLAY DEMO	A SOUND BY WAGNER
B BSOUND MAKER	A MUSIC NOTES WITH OCTAVE NOS.
A GREETINGS FROM W.V.A.C.C.	A ABE PIC.

THE FOLLOWING FILES CAN BE 'BRUN' OR YOU CAN 'BLOAD' THEM AND 'CALL' THE ADDRESS

B SOUND1(CALL2921)	B SOUND2(CALL3091)
B SOUND3(CALL2571)	B SOUND6(CALL3631)
B SOUND4(CALL3293)	B SOUND7(CALL5159)

Well this is the third issue for this column and it is obvious that the Editor is still short of contributions as he has accepted these ramblings without so much as a wimper. I usually leave them until the last minute before presenting the column to him and at such a late stage he is happy to accept anything. My request for some input for this column has fallen on deaf ears so you will have to put up with this lot.

Instead of using the common signs such as underline, dash and equals to define an area on printed text lets use the printer to do something a little more creative to produce a more impressive layout. The following lines indicate what I am on about.

```

===== -
===== I
===== 8
    
```

The fancy lines have been printed using a simple Print statement. Unfortunately most printer manuals don't seem to go into a lot of detail as to what their printers are capable of doing or maybe no one is interested in such trivia. Hopefully the following will be of benefit if you like to print lines.

These lines have been created using codes referred to in my manual as 'Line-Oriented Graphics'. I use an Apple DMP and parallel card so if you don't use the same equipment go and buy it - Apple dealers will love you - and you will be able to follow what I am on about. Or else it may be cheaper to look up your manual and figure it out yourself.

The lines are produced by sending a control code to your printer that causes it to repeat a single vertical dot pattern up to 9999 times horizontally. The code [ESC] "Vnnn"(ASCII character) is used. The 'nnnn' is a four digit number which specifies the length of the printed line. The following code will produce the first line in the group above.

```

30 PRINT CHR$(4);"PR#1"
40 PRINT CHR$(27);"V8150";"- "
50 PRINT CHR$(4);"PR#8"
    
```

It is not obvious in the above example without comparing the print to the second group of lines that in each set there is an extra line printed each time which is not controlled by the ASCII character. To stop this it is necessary to pass the data through a short machine language routine which deletes this line.

```

===== -
===== I
===== 8
    
```

The ASCII character printed at the end of the line identifies the character used to produce the pattern and has been added for clarity only. Try some other characters to get the result that suits your requirements.

To delete the extra unwanted row of dots the following code is required:

```

8388- A9 88
8382- 28 ED FD
    
```

Because the routine is so short it is easy enough to Poke it in using one Basic line.

```

28 POKE 768,169:POKE 769,8:
   POKE 770,32:POKE 771,237:
   POKE 772,253:POKE 773,96
    
```

To use this new method of printing, line 48 above would have to be changed to read as follows:

```

48 PRINT CHR$(27);"V8150";:POKE
   769,45:CALL 768
    
```

Note that the decimal value of the character is Poked into location 769 in line 48.

```

=====
=====
=====
    
```

B SOUND8(CALL2230)	B SOUND9(CALL5548)
B SOUND10(CALL5472)	B SOUND11(CALL4307)
B SOUND12(CALL5334)	B SOUND13(CALL4661)
B SOUND14(CALL4827)	B SOUND15(CALL3969)
B SOUND16(CALL4993)	

Volume # 019 - Disk 1 : FREE SECTORS = 0 (OK)

T ADDRESS INDEX *****	A ADDRESS INDEX.INSTRUCTIONS
A ADDRESS INDEX.LC	A BIRTH/ANNIVERSARY LIST
A CONVERT APPLEWRITER	T CRAE/AP *****
A CRAE/APA	A CRAE/APA.INSTRUCTIONS
I DISK SECTOR MAP	T DISPLAY ASC PAGE BY PAGE ****
A DISPLAY ASC PAGE BY PAGE.INST	B DISPLAY ASC PAGE BY PAGE.OBJ
A HELLO	T HUFFIM *****
A HUFFIN	A HUFFIN.INSTRUCTIONS
T JOYSTICK *****	B JOYSTICK WIRING DIAGRAM
T JOYSTICK.INFO	A JOYSTICK.INSTRUCTIONS
T LC-KILLER *****	A LC-KILLER.INSTRUCTIONS
B LC-KILLER.OBJ	A LOAN PROGRAM
T MEMAP (EXEC)	A PADDLE TEST
T POKE 33 *****	A POKE 33.INSTRUCTIONS
B POKE 33.OBJ	B POKE 33.OBJO
T POKE 33.SOURCE	T PUF *****
A PUF.INSTRUCTIONS	B PUF.OBJ
T PUF.SOURCE	A RAM TEST 48K
A SECTOR LIST	A TEXT TO APPLEWRITER
B TEXT TO APPLEWRITER.X	B TEXT.JOYSTICK.INFO
T TRACK 35 *****	A TRACK 35.INSTRUCTIONS
B TRACK 35.OBJ	B TRACK 35.OBJO
T TRACK 35.SOURCE	T UN-NEW *****
A UN-NEW.INSTRUCTIONS	B UN-NEW.OBJ
T UN-NEW.SOURCE	T X--- > DISK FULL

Volume # 020 - Disk 1 : FREE SECTORS = 0 (OK)

A ADDRESS SEARCH.INSTRUCTIONS	B ADDRESS SEARCH.OBJ
B ADDRESS SEARCH.OBJO	T ADDRESS SEARCH.SOURCE
T APPRINT	A APPRINT FILE MAKER
A BENCHMARK	A CANYON RUN COLOUR
A DISK FREE SPACE	A DISKSCAN
B DISKSCAN.OBJ1	B DISKSCAN.OBJ2
A ERROR HANDLER	B FAST.LOADER
B FAST.RWTS	A FASTBOOT CREATE
T GETFILE	A GETFILE INFO
B GETFILE.OBJ	T GETSECT
B GETSECT.OBJ	A HELLO
A HELLO CRAE/APA	A HYBRID HUFFIN
A LAPLACE INVERSION	A LAPLACE INVERSION.DOCUMENT
T LISTING	T LOWERCASE ASSEM LIST.TEXT
A MENU MAGIC	A MOONLANDING
A MORSE CODE	B MORSE CODE.OBJ
A PRETTYPRINT	A SHUFFLE
A SPACE CANYON	T X-> DISK FULL

[FEATURE ARTICLE]

Terc Basic Enhancements

by Sean Nolan

The TERC BASIC Enhancement package has been released into the public domain. The TERC BASIC Enhancements add over 50 additional BASIC commands to Applesoft which follow:

- * Mixing text and graphics anywhere on the screen.
- * Labeled subroutines with parameter passing and local variables.
- * Improved INPUT commands which simplify catching catching user errors.
- * IF/THEN/ELSE
- * Hi-Res SCRN and XPLOT commands.
- * Non-destructive pop-up menus and windows.
- * Debugging and diagnostic commands.
- * An optional 'compiler' (term used with blatant blatant inaccuracy) which allows you to write BASIC programs on a word processor.
- * No conflict with most Ampersand or USR libraries.
- * Expandability and documented entry points for machine language programmers.

```
*****
*                                     *
*   Product   *
*                                     *
*   Terc Basic *
*                                     *
*   Price     *
*   $5.00    *
*   p&p $1.00 *
*                                     *
*****
```

The TERC BASIC Enhancements were written for in-house use at Technical Education Research Centres (TERC), in Cambridge Massachusetts, by myself, Steve Beardslee, Steve Hunt, and Steve Bannasch. TERC is releasing the Enhancements into the public domain for non-commercial use only. The Enhancements may be ordered through Apple Q (with documentation on disk) or through myself (with printed documentation).

The Enhancements are ProDOS based, and thus require 64K of RAM. At present they are not compatible with the Laser 128 nor any other Apple clone. The core of the Enhancements use 6K of memory (\$800-\$1FFF). Some commands, such as the automatic MENU routine and pop-up windows, require more memory.

The BASIC Enhancements do not use Ampersand, but use the single quote (') instead. All Enhancements commands begin with this character. By avoiding the use of Ampersand, the Enhancements are able to add both new procedures and new functions to Applesoft (Ampersand lets you create procedures, but not functions). Procedures are commands which are meant to be executed at the beginning of a BASIC statement, such as the Applesoft commands HPlot, HOME, PRINT, and their ilk. Functions are called from within a numeric or string expression and return some value back to BASIC. Applesoft examples of functions are SCRN, PDL, RND, SIN, COS, and PEEK. (see quick reference sheet listing for all Enhancements commands).

Standard Goodies

The Enhancement commands can be separated loosely into four groups. First there are the standard goodies, commands which any self-respecting Applesoft extender should have. Next, the unusual commands, which make the TERC Enhancements unique. Third, the obscure technical commands. And then there are the commands which add no additional capabilities to BASIC but make it more readable.

The standard goodies include an ELSE to go along with IF and THEN, a rudimentary PRINT USING command for formatted numeric printing (yawn), and an improved INPUT anything routine. 'XPLOT draws Hi-Res lines in the complement of the background. A high resolution graphics SCRN function has been added, allowing you to read whether any Hi-Res pixel is on or off. A position-in-string function allows you to tell if any character or string is to be found within a second string. Three additional mathematical functions allow you to do genuine bitwise AND, OR, and XOR operations on numbers.

The two most important standard goodies, however, are its use of labeled subroutines and a Hi-Res character generator.

Labeled Subroutines

The Enhancements allow you to refer to subroutines by name rather than by line number. Instead of:

```
100 GOSUB 8045
```

which might call say, a help prompt subroutine, you can write

```
100 'GOSUB HELP
```

If you wanted to pass a particular prompt string to the HELP' subroutine, you could do it with

```
100 'GOSUB HELP [A$]
```

or

```
100 'GOSUB HELP ["Put the disk back and press a key."]
```

This is officially known as parameter passing. Along with the use of local variables (which the Enhancements support), this allows you to write structured code and use recursion. The latter is illustrated with a sample program on the enhancement disk. Called 'HILBERT', it draws a recursive curve. I translated it directly from the Pascal original (in the Call-A.P.P.L.E. JULY 1984 article "Hilbert and the Alligators") without having to change the recursive structure of the program.

Because of the ability to refer to subroutines by name instead of by line number, you can now write BASIC without the use of line numbers at all. At TERC, we write our programs with a word processor. The Enhancements come with a utility for translating word processor text files into runnable BASIC programs. You can, if you wish, not use it and write BASIC programs the normal way.

Mixing Text and Graphics

The Enhancements contain a full fledged Hi-Res character generator. This lets you put text characters anywhere on the Hi-Res display screen. Unlike many other Hi-Res character generators (yes, there are many of them), ours does not let you scroll text on the Hi-Res page. But there are compensating features: with the Enhancements you can draw characters at any pixel position you like, it is done like this:

```
'VTAB 11.5:'HTAB 31.2:PRINT A$
```

Most character generators restrict characters to a rigid 40 by 24 matrix. Our Character generator also supports underlining, variable character and line spacing, and three ways of combining the characters with their background.

The Enhancements don't provide Mac-like pull down menus, but do have their alternative. Pop-up windows containing text or menus can be placed anywhere on the screen; later they can be removed, leaving the background undamaged. One enhancement function, 'MENU, causes a menu in a pop up window to be displayed, interprets user keypresses, and returns with a number indicating which menu choice was chosen. You can even define a BASIC subroutine as a background process for 'MENU. As 'MENU waits for a keypress, the background subroutine will be called over and over. This gives the appearance of your BASIC program doing two things at once.

Our Sleazy Replacement for Virtual Memory

By now we've drifted into the second group of commands, those which make the TERC BASIC Enhancements unique. Pop-up windows, a Hi Res display, graphics buffers, and the Enhancements altogether take up a good deal of memory. Fortunately, ProDOS has added a CHAIN command to allow BASIC programs to be broken into pieces. The Enhancements go one step further with the 'DISK GOSUB command. This treats a BASIC program on disk as if it were a subroutine. For example, Program A might contain the statement:

```
'DISK GOSUB "SUBPROGRAM"
```

This runs the program SUBPROGRAM. When SUBPROGRAM ends, program A is automatically re-loaded, and resumes execution at the line following the 'DISK GOSUB statement. BASIC variables survive the tumultuous journey unscathed. If you've got a 128K machine, you can run all the pieces off the RAM disk with no noticeable delay.

Debugging Commands

The Enhancements contain built-in hex to decimal and decimal to hex converters. You can use them at the BASIC prompt or in your programs; this lets you use hex (base 16) instead of base 10 anywhere you like. For example:

```
HIMEM: '$9200 or Q = PEEK('$C000)
```

Are you tired? Depressed? Are your integer arrays being overwritten? Use 'STOP WHEN to find out where. 'STOP WHEN single steps through your BASIC program (at sub-glacial speed), continuously testing to see if a particular variable is being trashed, erased, or whatever. For example:

```
10 'STOP WHEN (W=7)
20 FOR W = 0 TO 10
30 PRINT W
40 NEXT W
```

The above program will crash with the message "BREAK IN 40" after the seventh trip through the FOR-NEXT loop.

'STOP WHEN is made possible by the way the Enhancements get control from Applesoft: we patched CHRGET. CHRGET is a little machine language subroutine that lives down in page zero. The BASIC interpreter calls it every time it needs to look at the next character in your BASIC program (for a full description of CHRGET, see Call A.P.P.L.E. March 1982). The patched CHRGET acts exactly as the original one does except that it gives the ENHANCEMENTS control whenever it parses a single quote (which transfers control to the Enhancements command parser) or a zero (which signals that the end of a BASIC line has been reached).

Input Trapping

Error trapping and bomb-proofing a program often requires as much attention as writing the real guts of the program. The Enhancements make error trapping easier in two ways - with improved INPUT and GET commands and with additional error trapping commands. The Enhancements INPUT routine allows you to specify ahead of time the maximum length of the string, allows all printable characters and only printable characters to be included (this allows commas, but not control characters), and tells you whether the user finished by pressing RETURN or by pressing ESCAPE. The new GET command takes a string parameter. It waits for the user to press one of the keys in the string, and returns the character position in the string of the matching character. For example:

```
A = 'GET FN ("QWERTY")
```

will wait for a "Q", "W", "E", "R", "T" OR "Y" to be pressed. If "Q" was pressed, A will be set to 1. If "W" was pressed, A will be set to 2, and so on. Any keypresses other than "Q", "W", "E", "R", "T" or "Y" will be ignored.

Applesoft Error trapping is improved three ways. First of all, the Enhancements have optional control-C trapping - it is not necessary to write an ONERR GOTO routine to trap these. ONERR routines, if used, can be referred to by label rather than by line number, and can be nested. This allows an error trapping routine to monitor errors in another error trapping routine - something I've actually needed to use when trapping the labyrinthine possibilities of desk access errors. The Enhancements also provide a third path to ONERR enlightenment. Normally, either of two things happen when an error occurs: either your program stops or it jumps to an error trapping routine. If you wish, you can cause errors to be simply ignored. This feature is useful in programs that save files to disk. If the file already exists, the traditional thing to do is install an error trapping routine and try to VERIFY the file. With the third path, you simply do the VERIFY and then check afterwards if it was successful.

Opening the Floodgates of Abuse

We now come to the Technical Enhancements commands. Most of these give you a better control over your BASIC variables. There is a function which returns the address of a variable, you can POKE at it or BSAVE it. You can create strings which point anywhere in memory, even into the softswitches. You can undimension an array. The Enhancements let you do lots of silly or dangerous things. Used foolishly, or with malice, they could scramble your Apple's memory.

The technical commands include some obscure trivia. There is a command, believe it or not, for retrieving half of the last random number seed as of the last ProDOS CHAIN executed. Why? Because we needed it at TERC to solve a particular problem. The Enhancements were developed for in-house use, and so contain only those commands which we needed. If the Enhancements had been written as a commercial product for sale, they would be less idiosyncratic. Instead of absurdities such as retrieving half of an old random number seed, there would be a command for drawing Hi-Res circles. It just so happened that we didn't need any circles. So you don't get circles... life is cruel.

The last group of Enhancement commands make BASIC more readable while adding no capabilities. 'DBL POKE, for example, POKES a 16-bit word into memory. You can do it without the enhancements with two regular POKE statements. 'PI replaces 3.14159. 'BELL replaces PRINT CHR\$(7), and so on. Need I go on?

How to Get the Enhancements

I've covered the major Enhancements commands. Any commands which I haven't discussed are either self-explanatory or rarely used. As one of the Enhancements programmers, I can not remain impartial; yet I feel that the Enhancements, at either \$5.00 or \$US15.00 (take your pick) is an unbeatable bargain.

The Enhancements are available from APPLE-Q for five dollars. It comes with documentation on disk as ProDOS text files. The Enhancements are also available from yours truly (68 Roseland St., Somerville Mass. 02143) for \$US15.00, including a 50-page printed manual and one free upgrade disk (which will be sent to you later, or maybe never - how's that for a guarantee?). Future upgrades may include Double-Hi-Res graphics support or Laser 128 compatibility.

[Published in the June, 1987 issue of Call-A.P.P.L.E.]

[Programing Languages]

by: Mark Ritchie.

Not a lot of people know much about other languages other than that they exist and they might be behind their favourite arcade game. This month I have written an article on a programing language called 'C'.

You might call C a 'middle class' language, since it is easier to write and comprehend than machine language, but it lacks some of the features of higher languages. C was created by the Bell Lab's in the early 1970's where it was first used to transport the UNIX operating system from a DEC minicomputer to other computers.

Today, C is used to write everything from arcade games to business software. It is a favourite of commercial software programmers for a couple of reasons.

- A) C programs compare favourably with machine language programs in performance.
- B) C is not tied to any particular operating system, making it suitable for a variety of computer systems. This makes it suitable for programs that must be translated for several different computers.
- C) There are also a number of reasonably priced compilers that have made their way into the home computer market. Due to this reason, C has become popular with hobby programmers, too.

As far as keywords go, C has fewer than 30, and about half of them are used to declare variables and other data objects. The most important keywords are, else, cas, switch, default, for, do, while, break, continue, return, goto

You might ask how useful a language is, that has so few words but unlike BASIC which has 50-200 keywords, depending on the version. The answer lies in the fact that C never uses two words where one will do. For example, where BASIC uses FOR and NEXT to create a FOR loop, C dispenses with NEXT. The occurrence of 'for, while, or do' is sufficient to identify a loop in C. Another reason for few keywords is that one can have multiple meanings, depending on where it is placed in a program. For instance, 'int' means 'I plan to use an integer variable with the following name,' if used in variable declaration. When it occurs before a function, it means 'The following function returns an integer value.'

Another powerful data object is the pointer, a variable that points to some other variable. A common use for it is to access individual elements of an array. They can also be used to include direct manipulation of the computer's memory.

An array in C is a collection of data objects of the same type under one name. In BASIC, it would be the same as, NAME\$(1) and NAME\$(2). C data types may be combined quite freely, allowing you to create arrays of pointers, pointers which point at pointers, arrays of structures, pointers to structures, self-referential structures, and so on.

The C language is easily trapable. It does not provide input/output (I/O) functions in the language itself but every compiler includes a so called standard library of I/O functions to do tasks such as printing to the screen, reading and writing files and so on. A program that uses nothing but standard I/O functions should be transportable, with little changes, from one computer to another for which there is a C compiler. However, this is not usually the case, except for very plain programs, since most programmers will take advantage of special features unique to the host computer.

ooooo000ooooo

[Advertisements]

FOR SALE -> Contact: Graham Black at the Trading Table or ph.(07) 883-1525
 ~~~~~

|                                                                |              |
|----------------------------------------------------------------|--------------|
| Speed Demon (speeds up the apple 3.6 times faster than normal) | \$ 500       |
| Roland DXY-800 Plotter (8 colour) includes many pens and paper | \$1050       |
| Alf Synthesizer + amplifier + approx. 30 disk of music .....   | \$ 400       |
| DIGITEK Serial Interface DK 24-4 .....                         | \$ 75        |
| //e Mouse .....                                                | \$ 225       |
| //e numeric keypad .....                                       | \$ 75        |
| PASCAL Reference manuals .....                                 | \$ offers    |
| Microfazer 64k printer buffer. serial in/parallel out .....    | \$ 195       |
| MCP printer/plotter (plus pens) .....                          | \$ 90        |
| APPLE /// Softcard III by MICROSOFT (2 only) .....             | \$ 300       |
| Assorted APPLE /// software (incl. Visicalc III) .....         | \$ offers    |
| Price-tag gun (2 only) .....                                   | \$ offers    |
| Telex rolls .....                                              | \$ 4         |
| ZARDAX (DOS 3.3) .....                                         | \$ 165       |
| ZARDAX II (ProDOS) .....                                       | \$ 165       |
| <u>RS-232 Inline Switch Box. (male to female) .....</u>        | <u>\$ 20</u> |

**FOR SALE -> Contact: Vince Crosdale at the Trading Table or ph.(07) 351-3090**  
 ~~~~~

Enhanced Apple //e 128k, two disk drives, monitor, parallel card, RGB card, manuals many disks included. \$2000 ph.(07) 351 3090

FOR SALE -> Contact Bob Karlsson at the club or phone (07) 208 7855
 ~~~~~

16k Ram card for Apple ][+ (increase RAM to 64k)  
 PAL colour card  
 Make an offer.

[ TRADING TABLE ]

During a recent discussion, it has come to our attention that a number of the members are under the wrong impression when it comes to the profits made from the Trading Table. Apple-Q owns and operates the Trading Table. ALL PROFITS go towards the routine maintenance of the club and towards maintaining the software bank and the library. Apart from membership fees, this is the only source of income for the Club and therefore a vital key to our continued operation. The profits from each meeting help offset the cost of producing the newsletter, purchasing disks, purchasing of software and magazines for the libraries, paying the rent and the cleaning of the club rooms. Items offered for sale at the Trading Table are initially purchased from the cheapest source available AT THE CURRENT TIME, and a small percentage of the original purchase price is added to the resale price to allow the Club to keep its head above water, so to speak. We offer this service as a convenience to members, and it is only through the members that the Club can continue to operate in its present format. If anyone has a source of supply for items for the Trading Table that would appear to be cheaper than our present source then please let us know, as it is you, the members of Apple Q, that benefit from this operation.

PRICE LIST - as at: 2nd AUGUST 1987

| <u>Description Details</u>                   | <u>Price</u> |
|----------------------------------------------|--------------|
| 5.25in disks (singles)                       | 2.50         |
| 5.25 DS/DD DISKS (PKT.10)                    | 12.00        |
| 3.5inch SS/DD DISKS (per)                    | 4.00         |
| BASIC PROGRAMMING (book)                     | 15.00        |
| WORKSHOPS VOL.1 (book)                       | 16.50        |
| WORKSHOPS VOL.2 (book)                       | 16.50        |
| COMPUTER CONNECTION (book)                   | 10.00        |
| WP WITH ZARDAX (book)                        | 15.00        |
| EPSON LX/GX-80 RIBBON CARTRIDGE              | 9.25         |
| EPSON MX80 RIBBON.CARTRIDGE                  | 11.00        |
| DISK DRAWERS 5.25in (100)                    | 15.00        |
| FD30 MINI DISK TRAY (30)                     | 9.90         |
| DISK LABELS.FanFold (1000pk)                 | 20.00        |
| 6-RING BINDER.HARD COPY                      | 7.50         |
| MAILING LABELS fanfold (12 sheets of 8)      | 2.00         |
| ZERO-FORCE SOCKETS I/O                       | 27.00        |
| MISC.SOFT/HARDWARE                           | .50          |
| ASSORTED BOOKS                               | .50          |
| 11 x 9.5in. cleancut paper (box 2000 sheets) | 42.00        |
| EPSON LX-800 PRINTER                         | 510.00       |

*The above prices are correct at the time of printing and are subject change without notice.*

It is not always possible to have every single item in the price list available at each meeting, so if there is anything in particular that you would like to see, then phone me on (07) 883-1525 after mid-day, at least two working days before the meeting and I will attempt to get it for you. Also you can place an order for any of the above items along with your deposit of at least 25%, at the meeting.

We will have CREDIT CARD facilities available in the very near future.

Bankcard - Mastercard - Visa.