THE R O M NEWSLETTER
USERS GROUP OF ORANGE COUNTY
17301 SANTA ISABEL STREET
FOUNTAIN VALLEY, CA 92708

XXX
Dallas TI Computer Group (DTIHC
PO Box 29863
Dallas,
TX 75229

JAN 1989

SERVING THE TI 99/4A HOME COMPUTER COMMUNITY

WE MEET AT MERCURY

TIME AND PLACE OF MEETING
The <u>FIRST</u> Thursday of each month at

MERCURY SAVINGS and LOAN
7:30 PM

West of Beach at 7813 Edinger Ave., Huntington Beach, Cal. Use the WEST enterance. Park on the west side of the building. All are welcome.

U.G.O.C. OFFICERS

PRESIDENT	JIM SWEDLOW	897-9209
VICE-PRES	JIM JOLLY	531-7782
SECRETARY	RICHARD ATKINS	637-7410
TREASURER	JERRY RASH	631-0579
PAST-PRES	ROBERT HARPER	744-2517

COMMITTEE CHAIRMEN

JIM MORRIS	MEMBERSHIP	546-8354
SILES BAZERMAN	NEWSLETTER	897-2868
BEN HATHEWAY	BULLETIN BOARD	751-4332

SOFTWARE LIBRARY

KNUTE ERSLAND.....................842-0859

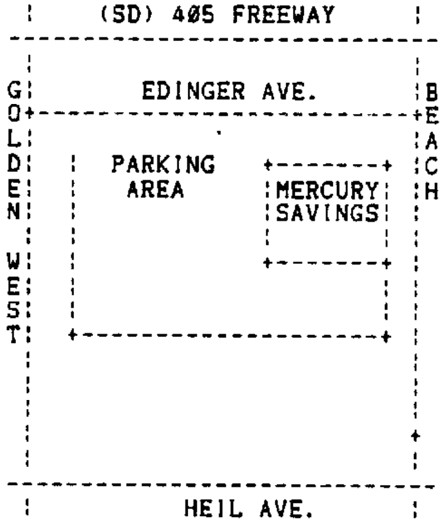
NEWSLETTER LIBRARY

EARL RAGUSE....(Phone for time)...847-5875

PUBLICATION NOTICE

The ROM is published by the Home Computer Users Group of Orange County, a non-profit organization. Permission is granted to similar groups to reproduce articles published in the ROM provided author and source credits are included. Opinions expressed in this publication are those of the author and do not necessarily represent those of the ROM, it's Editor, or the HCUGOC, it's officers or members.

We solicit letters and articles of interest to the TI-99/4A user community. Material accepted may be edited for fit and format. No payment is offered nor intended (other than your byline).



NEWSLETTER CONTRIBUTORS

NEWT ARMSTRONG	EDITOR EMERITUS
EARL RAGUSE	PRODUCTION
EARL RAGUSE	TI FORTH
ADRIAN ROBINSON	ASSEMBLY
ROBERT SHAFFER	CIRCULATION
JIM SWEDLOW	AT LARGE

TI CLUB ACTIVITIES

CLUB	ACTION	DATE	INFO
UGOC	GENERAL MEETING	FEB 02	897-9209
BUG	GENERAL MEETING	FEB 06	871-3405
UGOC	LIBRARY, FTNVLY	FEB 06	842-0859
UGOC	ASSEMBLY SIG	FEB 09	537-1839
ET	GENERAL MEETING	FEB 11	837-8757
UGOC	LIBRARY, FTNVLY	FEB 13	842-0859
UGOC	BOARD MEETING	FEB 16	897-9209
UGOC	ASSEMBLY SIG	FEB 23	537-1839
UGOC	NSLETTER LIBRARY	CALL	847-5875

FROM THE PRESIDENT

By Jim Swedlow

FEBRUARY MEETING

We have a couple of interesting things coming in February. We will have a speaker (name and subject are still "under discussion").

We will also have a demonstration of the GENIAL TRAVelER. This is a "diskazine" or a TI based magazine on disk published by Barry Traver. UGOC has obtain permission to resell the GT (as it is known) at a rate just above the subscription cost. Volume 2, Numbers 1, 2 and 3 will be available.

TI FEST WEST

Last reminder: the Fest West is February 18th and 19th from 9 to 6 at the Clarion Hotel, 2223 El Cajon Blvd in San Diego. Admission is $4 at the door. This looks to be another great TI event. See you there.

NEWSLETTER LIBRARY

Have you ever had a problem and just could not find the answer anywhere? The odds are that it is in our newsletter library. We have newsletters from over fifty other user groups. They contain a wealth of information.

Call Earl Raguse at 847-5875 to make an appointment. You will learn something new and useful from this collection.

SOFTWARE OF THE YEAR

Your board is considering starting an annual recognition for the software item of the year. This would be another way for us to help support those who expend their creative energies for the 4A. Any comments would be welcome.

LIBRARY UPDATE

PLUS! (FW68): A collection of hints and utilities for the FUNNELWEB user.

EE BONDMASTER (FW6Ø): Now reflects US Bond interest rates thru November, 1988. To get the current interest rates, call 1-8ØØ-US-BONDS.

TELCO (FW58 and FW59): Version 2.3.

Look for some 1989 calendars in the next release.

'Nuff said.

UGOC BOARD MEETING MINUTES 12/15/88

By Earl Raguse, (sub for Richard Atkins)

Jim Swedlow, President called meeting to order at 7:4Ø at Ben Hatheways home.

Jim reports that a Thank You letter has been sent to Mercury Savings. We are in the process of moving our checking account to Mercury.

Jerry Rash, Treasurer reports that we are in good financial shape again. It was agreed that the last 5Ø disks from our bulk purchase would be made available to the library.

Jim Morris, Membership reports that we still have 8Ø members. Nine of these currently are up for renewal, and 12 are overdue. We acquired 1Ø new members since 1/1/88. It was decided that we should list all new members and renewals in the ROM.

Siles Bazerman, Editor has has turned over the latest ROM to Earl Raguse ready for printing. Siles is still looking for articles from the membership on Basic, XBasic, and TIW/TIB. A regular contributor is needed but we will appreciate single articles if you write one. If you can't write about your big successes because you haven't any, then write about your troubles, but write.

Ben Hatheway, BBS Sysop says the board is getting better and busier all the time. Xmodem has been speeded up. The San Diego BBS may begin using our system.

Earl Raguse, Paper Library says everything is now bound in three ring binders, and the library is open almost any time, just call to make sure.

January Meeting: we will have a hardware demo. An 8Ø column card, a RAM disk, and a PGRAM will be shown. The results of the survey which will probably be distributed in the December meeting may be available.

February Meeting: we plan on having Jim Swedlow talk about TIW Headers and Footers, and Newt Armstrong will demo SuperSpace.

We will be raffling a TymShare 12ØØ baud modem in either Jan or Feb. A discussion was held as to what we could do to encourage software writers. Stan Corbin suggested, and we will follow through, that we recognize the best programs and programmmers, in a public way.

Meeting adjourned at 9:14

ASSEMBLY LANGUAGE
by Adrian Robinson

It seems we have some readers who are not dedicated Assembly programmers but who look for an occasional tidbit that can be called from an XBasic program. A frequent requirement in any language is to sort a string array(list). Since sorting is an inherently slow process it is nice to be able to take advantage of the speed of Assembly Language. However not all sort algorithms are alike. I have in the past compared a variety of sorts ranging from the very slow Bubble Sort to the very fast such as Quicksort and Shell-Metzer sorts. For very short lists less than about 30 elements, there is little to choose among them all. None take much time in Assembly. However, as the lists grow, the slow algorithms fall rapidly behind. Up to several hundred elements, the Shell-Metzer and Quicksort are about equal in speed and for very large lists the Quicksort pulls ahead. There is, though, a disadvantage in the Qicksort in that it makes use of pointer stacks, which require more and more memory as lists grow, while Shell-Metzer does not use such pointers. In addition Shell-Metzer is short and easy to code, as can be seen from the following XBasic listing:

```
10 P=N
20 P=INT(P/2) :: IF P=0 THEN
 RETURN
30 FOR I=1 TO N-P
40 FOR J=I TO 1 STEP -P
50 IF A$(J)<=A$(J+P) THEN 70
60 T$=A$(J) :: A$(J)=A$(J+P)
 :: A$(J+P)=T$ :: NEXT J
70 NEXT I :: GOTO 20
```

I consider the Shell-Metzer to be a good general purpose sort algorithm and have provided an Assembly listing below. I have included an additional feature in that the caller may specify that the sort be either ascending or descending.

The array to be sorted is constructed in your XBasic program and the sort is then accessed through:

```
CALL LINK("SORT",N,K$(),X)
```

where N is the number of elements in the array and K$ is the array name. If X=0 then the sort will be in ascending order and if X has any non-zero value then the sort will be descending. The variable names need not be those specified here, only their type and position in the LINK list are important. The program listing comments include statements similar to those in the XBasic routine above as an aid to following the program logic. It may be noted that, following execution of STRREF, I use the string length to clearly delimit each string with a zero byte obtained from the high byte of the index register. That comment was thrown in for the committed Assembly people who may be reading.

```
*****************************************
* Shell-Metzner Sort Routine            *
*     by Adrian Robinson                *
* Call from Extended Basic:             *
* CALL LINK("SORT",N,K$(),X)            *
* With N = Number of elements in Array  *
*     K$ = Name of String Array         *
*      X = Order of Sort:               *
*          0 = Ascending Order          *
*      Not 0 = Descending Order         *
*****************************************
*
       DEF  SORT
NUMREF EQU  >200C
STRASG EQU  >2010
STRREF EQU  >2014
XMLLNK EQU  >2018
FAC    EQU  >834A
STATUS EQU  >837C
GPLWS  EQU  >83E0
MYWS   BSS  32
SAV11  BSS  2
BUF1   BSS  256    String Compare Buffer
BUF2   BSS  256      "       "      "
FF     BYTE >FF    Maximum String Length
       EVEN
*
SORT   MOV  R11,@SAV11
       LWPI MYWS
       CLR  R0                Not an Array
       LI   R1,3           Third Parameter
       BLWP @NUMREF Get X from CALL LINK
       BLWP @XMLLNK                Convert
       DATA >12B8  Floating Point to Int
       MOV  @FAC,R10
       LI   R1,1           First Parameter
       BLWP @NUMREF Get N from CALL LINK
       BLWP @XMLLNK                Convert
       DATA >12B8  Floating Point to Int
       MOV  @FAC,R9
*
       LI   R1,2    Second param in LINK
       MOV  R9,R8                      P=N
LOOP   SRL  R8,1              P=INT(P/2)
       JEQ  RETURN    IF P=0 THEN RETURN
       LI   R3,1
ILUP   MOV  R3,R4           FOR I=1 TO N-P
JLUP   MOV  R4,R0   FOR J=I TO 1 STEP -P
       LI   R2,BUF1     Get K$(J) to BUF1
       MOVB @FF,@BUF1      Maximum Length
       BLWP @STRREF
       MOVB @BUF1,R6        Actual Length
       SRL  R6,8
       MOVB R6,@BUF1+1(R6)  Delimiter  0
       MOV  R4,R7
       A    R8,R7
       MOV  R7,R0
       LI   R2,BUF2  Get K$(J+P) to BUF2
       MOVB @FF,@BUF2      Maximum Length
       BLWP @STRREF
       MOVB @BUF2,R5        Actual Length
       SRL  R5,8
       MOVB R5,@BUF2+1(R5)  Delimiter  0
       C    R5,R6
       JH   $+4
       MOV  R6,R5
       CLR  R6
COMP   INC  R6           Compare BUF1,BUF2
       CB   @BUF1(R6),@BUF2(R6)
       JGT  ORD1          IF K$(J)>K$(J+P)
       JLT  ORD2          IF K$(J)<K$(J+P)
       C    R6,R5
       JLT  COMP
       JMP  NEXTI       . IF K$(J)=K$(J+P)
ORD1   MOV  R10,R10          Order Flag=0?
       JEQ  SWAP
       JMP  NEXTI
ORD2   MOV  R10,R10          Order Flag=0?
       JEQ  NEXTI
SWAP   MOV  R7,R0   K$(J+P)=K$(J) (BUF1)
       LI   R2,BUF1
       BLWP @STRASG
       MOV  R4,R0   K$(J)=K$(J+P) (BUF2)
       LI   R2,BUF2
       BLWP @STRASG
       S    R8,R4          J=J-P  (STEP -P)
       JGT  JLUP                    NEXT J
NEXTI  INC  R3
       MOV  R9,R7
       S    R8,R7
       C    R3,R7
       JLE  ILUP     IF I<=N-P THEN ILUP
       JMP  LOOP                 ELSE LOOP
*
RETURN LWPI GPLWS
       MOV  @SAV11,R11
       CLR  @STATUS
       RT
       END
```

=======================================

MEMBERSHIP CORNER
By Jim Morris

Membership is currently at seventy four.. Six members were dropped as they now have other interests. In addition six members are on ninety day hold and fifteen memberships are currently due. During the year we lost nineteen members and acquired twelve. As you can see we do have an attrition problem and eventually it could become a major problem for all of us. So please if at all possible invite several of your friends to attend one of our club meetings so that they too can become a computer hacker. Current membership lists and updated library lists are available upon request. Extra copies of ROMS as well as technical data extracted from other club newsletters are also available. Two newe prospective members are expected to attend the January meeting.

TI BITS * Number 24
By Jim Swedlow

EA DISK ERROR CODES

Some programs (like Archiver) display a number when they encounter a disk error (something like "IO ERROR #7"). The numbers by themselves are of little use. Here is what they mean:

Ø UNKNOWN DEVICE - Could not find the specified drive.

1 WRITE PROTECTED - The disk is write protected.

2 BAD OPEN ATTRIBUTE - One or more OPEN options were illegal or didn't match the file characteristics.

3 ILLEGAL OPERATION - The book says that this code should not be generated!

4 OUT OF SPACE - The disk is full or you are trying to open more files than are allowed (127).

5 END OF FILE - You are trying to read beyond the end of the file.

6 DEVICE ERROR - The disk is not initialized, the disk is damaged, the disk drive is broken (oh no!), the drive door is open, etc.

7 FILE ERROR - The file doesn't exist or you are trying to read a BASIC file as if it were data.

These are the same as the second digit in the BASIC disk error codes.

MAGIC NINES

A fun program by Jim Peterson of the TIGER CLUB.

```
1ØØ ! MAGIC NINES
       By Jim Peterson
11Ø CALL CLEAR
12Ø PRINT ::
    PRINT "ENTER ANY 3 DIGIT NUMBER":
    "WITH 3 DIFFERENT DIGITS"
13Ø INPUT N ::
    PRINT ::
    IF N<>INT(N) OR N>999 OR N<1ØØ
    THEN 12Ø ELSE N$=STR$(N)
14Ø IF SEG$(N$,1,1)=SEG$(N$,2,1) OR
    SEG$(N$,1,1)=SEG$(N$,3,1) OR
    SEG$(N$,2,1)=SEG$(N$,3,1) THEN
    PRINT ">USE 3 DIFFERENT DIGITS<" ::
    GOTO 12Ø
15Ø N2$,N4$="" ::
    FOR J=1 TO 3 ::
    N2$=SEG$(N$,J,1)&N2$ ::
    NEXT J
16Ø N2=VAL(N2$) ::
    N3=ABS(N-N2)
17Ø PRINT N$;" BACKWARDS IS ";N2$ ::
    PRINT
18Ø N3$=STR$(N3) ::
    IF N3<1ØØ THEN N3$="Ø"&N3$
19Ø IF N>N2 THEN PRINT
    N$;" MINUS ";N2$;" IS ";N3$
    ELSE PRINT
    N2$;" MINUS ";N$;" IS ";N3$
2ØØ PRINT ::
    FOR J=1 TO 3 ::
    N4$=SEG$(N3$,J,1)&N4$ ::
    NEXT J
21Ø PRINT N3$;" BACKWARDS IS ";N4$: :
    N3$;" PLUS  ";N4$;" IS 1Ø89": :
22Ø PRINT :"I KNEW THAT WOULD BE": :
    "THE ANSWER.  LIST THE": :
    "PROGRAM AND SEE!"
23Ø !!!!!!!!!!!!!!!!!!!!!
24Ø !   THE ANSWER IS   !
25Ø !       1Ø89        !
26Ø !!!!!!!!!!!!!!!!!!!!!
```

TI LIVES

The December, 1988 issue of PC Computing (a magazine normally dedicated to MS DOS machines) has an article entitled "Gone But Not Forgotten". There is coverage of most of the orphans (TI, Osborne, Eagle, PCjr, etc). By the time you read this, you won't be able to buy that issue, but it might be worth checking out in your library. A couple of interesting quotes:

"The 99/4A and PCjr were early experiments in the home computing market. They weren't nearly as fast or powerful as other computers of their time, yet in many homes they continue to fulfill the role that visionaries once predicted for them: they've evolved from somber, cold pieces of machinery to tools that are useful and fun."

"Fans of the TI 99/4A are legion, and they form a network of users that's as lively as FOG, if less structured. TI user groups number 3ØØ with the Chicago group the largest at nearly 55Ø members. Other TI organizations are small but spirited. The 14 member north New Jersey group works with other New York area computer groups to sponsor the TI Computer Fest which each year draws about 3ØØ people to attend workshops and hear speakers."

[FOG is an international User Group for owners of CPM and MS DOS machines - Ed]

TI WRITER MARGINS

For the first item in this column, I used a device called a hanging indent. It looks like this:

CONTINUED Page 8

BASICALLY BASIC
by N. Armstrong

First, a correction for last month. The Formatter clobbered my program listing in statements 18Ø and 19Ø. They should have been:

```
18Ø N$=N$&".Ø"
19Ø N$=N$&"Ø"
```

This is called concatenating and builds up N$ by appending whatever is in quotes to the end of the variable.

Now, back to this month's offering.

An uncluttered, easily understood screen is a big help in making a program "User Friendly". The following program keeps a running total for a checkbook. Key it in, RUN it, and let me know if you find it User Friendly.

```
1Ø5 REM ********************
*       INITIALIZATION     *
****************************
11Ø CALL CLEAR
12Ø CALL SCREEN(5)
13Ø FOR I=1 TO 8
14Ø CALL COLOR(I,5,1)
15Ø NEXT I
16Ø FOR I=9 TO 12
17Ø CALL COLOR(I,15,1)
18Ø NEXT I
19Ø REM ********************
*        TITLE SCREEN      *
****************************
2ØØ PRINT "        AN ACCUMUL
ATOR": :
21Ø PRINT "FOR CHECKBOOK RUN
NING TOTALS": : : : : : : :
: : :
22Ø PRINT "A TUTORIAL FROM

                 THE ROM NEWS
LETTER": :
23Ø PRINT "             BY N
. ARMSTRONG";
24Ø CALL SCREEN(16)
25Ø FOR I=1 TO 15ØØ
26Ø NEXT I
27Ø CALL CLEAR
275 REM ********************
*        PROGRAM START     *
****************************
28Ø INPUT "BEGINING BALANCE
$":TP
29Ø T=T+TP
3ØØ GOSUB 6ØØ
31Ø PRINT
32Ø A$="Check Deposit Atm or
 Quit"
325 REM ********************
*        MAIN LOOP START   *
****************************
33Ø FOR D=1 TO LEN(A$)
34Ø S=ASC(SEG$(A$,D,1))
35Ø CALL HCHAR(24,D+2,S)
36Ø NEXT D
365 REM ********************
*       CURSER CHANGE      *
****************************
37Ø S=32+2*(S=32)
38Ø CALL HCHAR(24,3Ø,S)
39Ø FOR D=1 TO 5
4ØØ CALL KEY(3,K,K)
41Ø IF (K=65)+(K=67)+(K=68)+
(K=81)THEN 44Ø
42Ø NEXT D
43Ø GOTO 37Ø
435 REM ********************
*       ROUTINE SELECT     *
****************************
44Ø ON -(K=65)-2*(K=67)-3*(K
=68)-4*(K=81)GOTO 46Ø,48Ø,51
Ø,45Ø
445 REM ********************
*         STOP  ROUTINE    *
****************************
45Ø STOP
455 REM ********************
*          ATM  ROUTINE    *
****************************
46Ø INPUT "ATM AMOUNT
 ":TP
47Ø GOTO 49Ø
475 REM ********************
*        CHECK ROUTINE     *
****************************
48Ø INPUT "CHECK AMOUNT
 ":TP
49Ø T=T-TP
5ØØ GOTO 53Ø

5Ø5 REM ********************
*      DEPOSIT ROUTINE     *
****************************
51Ø INPUT "DEPOSIT AMOUNT
 ":TP
52Ø T=T+TP
53Ø GOSUB 6ØØ
535 REM ********************
*       TOTAL  ROUTINE     *
****************************
54Ø PRINT "TOTAL
$"
55Ø TP=T
56Ø GOSUB 6ØØ
57Ø PRINT
58Ø GOTO 33Ø
59Ø REM ********************
*        DP SUBROUTINE     *
****************************
6ØØ T$=STR$(TP)
61Ø P=POS(T$,".",1)
62Ø IF P>Ø THEN 65Ø
63Ø T$=T$&".ØØ"
64Ø GOTO 61Ø
65Ø IF P=(LEN(T$))-2 THEN 67
Ø
66Ø T$=T$&"Ø"
67Ø CALL HCHAR(23,21,32,9)
68Ø FOR D=1 TO LEN(T$)
69Ø S=ASC(SEG$(T$,D,1))
7ØØ CALL HCHAR(23,(26-P)+D,S
)
71Ø NEXT D
72Ø RETURN
73Ø END
```

AND SO FORTH#39

By Earl Raguse

FOOLING FORTH

Some times we need to sort of bend the rules to make things easier. Forth, like all other computer languages has rules which are not ment to be violated. However, Forth has the unique capability that one can change the rules that one doesn't like, or that may be inconvenient at the moment. In some cases it seems very inconvenient to live by the Forth rule that a word must be defined before it can be used in a definition.

My first bout with this was when I tried to write a 2 page DIRectory word. I first introduced DIRectory (1 page), in ASF#21 (Aug 87 ROM), then again in ASF#25 (Dec 87 ROM), in which I used vectoring to solve the 2 page DIR problem. CASE OF ENDOF and ENDCASE which are the crux of DIR were fully explained in ASF#21, and I can't take the space to repeat all that, but I will summarize. For those of you who don't have the above ROMs, see the offer at the end of this column.

Partial Screens #6, #7, #31, and #32 show how I solved the problem by defining an empty variable POINTER, so that I could use it in the definition of PAGE2, then after DIR2 was defined on Screen #31, I loaded the address of DIR2 (' finds it), into POINTER, then later on boot up, Screen #3 executes DIR which calls PAGE2, (Screen #6) which in turn looks in POINTER for the address, actually the Parameter Field Address (PFA), converts it to the Code Field Address (CFA), then EXECUTEs it, to do the work. This is called vectoring and is a form of indirect addressing.

Why did I take this round about approach? Why not just have written DIR2 into the definition of DIR? This is not allowed because DIR2 did not yet exist. Well then, why not write DIR2 first, then DIR. The old chicken/egg problem, because then DIR doesn't exist. As you can see, my solution may be a little convoluted, but I worked within the rules of Forth and got the job done.

Now it happens that there is a rather shrewd Canadian programmer by the name of Michal Jaggerman who didn't want to abide by that rule either, so he wrote two words to bend the rules a little, this allowed him to postpone defining a word that he wanted to use right away. His clever words DEFER and IS were recently brought to my attention by Lutz Winkler of San Diego. They do essentially what I did, but its done in a straight forward way, and is easy to understand and use. The difficulty with this approach, if there is one, is that you must have previously defined the words DEFER and IS, and have them in your dictionary.

Michal's DEFER and IS are shown on Screen #83 along with an example of their use. I'm sure you won't have any trouble replacing my scheme with his. Just DEFER PAGE2 instead of creating variable POINTER and defining PAGE2. Then at the end of Screen #31 just replace ' DIR2 POINTER ! with ' DIR2 IS PAGE2.

For those of you who do not have the above issues of the ROM, if you use a little soft soap on our Editor, plus a standard $2.ØØ library fee, he will provide you with a disk copy of almost any ROM; there are some limited exceptions. Where that applies, see me, as the Paper Librarian, I'm sure I can find that issue in our Library. You may borrow and copy that. Also, if you apply the above softsoap on me, with the standard Library fee, I will provide you with a disk copy of the ASF article in question, along with 4 or 5 others. All ASF articles are available on disk. The above offer applies not only to our regular members, but also to the Editors of any any TI UG that we exchange newsletters with. Please include a stamped self addressed disk mailer with your request or an additional $1 to cover costs.

Also the Forth disk that uses the above 2 page DIR has a lot of nice stuff, including the muchly improved Editor, that doesn't show in the above example, it is intended for people just starting Forth as well as dyed in the wool experienced users.

This is my Forth disk number four which was created on the Fourth of July (1988), and is available hence-forth for free for all who apply for it in a forth-right manner with a blank disk.

C U next time, May the FORTH be with U.

==

REDFACE DEPT

The formatter bug bit Newt, and the improper use of the RES command bit me. The proper lines for LOTTO are below.

```
21Ø IF Y(I)>49 OR Y(I)<1 THE
N 2ØØ
23Ø IF Y(I)=Y(F)THEN 2ØØ
```

AND SO FORTH SUBJECTS IN THE ROM

Here is a list of "all" AND SO FORTH articles printed in the ROM over the past three years. All are available from the UGOC Paper Library.

ASF#	SUBJECT	ROM
1	Getting Started.....	Nov 1985
2	UFW's + Sprites.....	Dec 1985
3	Sprites + GSTR$.....	Jan 1986
4	CSTR$ + Disk Dir....	Feb 1986
5	Dspl Cont + Prt Cont	Mar 1986
6	Draw Graph/no labels	May 1986
7	Graphics Demo.......	Jun 1986
8	Font Control........	Jul 1986
9	Trig + more Graphics	Aug 1986
1Ø	RKEY,Prmpt + Music..	Sep 1986
11	More Music..........	Oct 1986
12	OOPS! Abbr etc......	Nov 1986
13	Hammurabi!..........	Dec 1986
14	Forth Talk..........	Jan 1987
15	String Handling/UFWs	Feb 1987
16	File Access.........	Mar 1987
17	String Sorting......	Apr 1987
18	More Sorting........	May 1987
19	Music w/chords......	Jun 1987
2Ø	Music w/neg dur, PNn	Jul 1987
21	Misc stuff/dir/UFWs.	Aug 1987
22	Starting Again	Sep 1987
23	Mike Kabala sound...	Oct 1987
24	DUMP,<BUILDS, etc...	Nov 1987
25	Menu,vectoring,array	Dec 1987
26	Potpurri,Reverse Vid	Jan 1988
27	Stack manipulators..	Feb 1988
28	1, 2, 3 Dim Arrays..	Mar 1988
29	Statisical Rounding.	Apr 1988
3Ø	Variable Names......	May 1988
31	Bases and Binary....	Jun 1988
32	Ed/Append, Scr Title	Jul 1988
33	Number Formating....	Aug 1988
34	CLINE/Graphics......	Sep 1988
35	HOTPO and WHILE.....	Oct 1988
36	Dbl Precision HOTPO.	Sep 1988
37	F83/TI and Dbl Prec	Nov 1988
38	More Dbl Precision	Dec 1988
39	DEFER / IS DIR/DIR2	Jan 1988

```
SCR #6
 Ø  ( MENU SELECT 1 )  DECIMAL
 1  Ø VARIABLE POINTER 4 ALLOT
 2  : PAGE2 POINTER @@ CFA EXECUTE ;
 3  : DIR 7 LOAD 28 2Ø AT KEY  48 - CASE
 4  -16 OF     PAGE2               ENDOF
 5    Ø OF     ABORT               ENDOF

13    8 OF 84 LOAD                 ENDOF
14    9 OF 8Ø LOAD                 ENDOF
15    ENDCASE ;

SCR #7
 Ø  ( MENU SELECTION EGR 7/4/88) HOME
 1  ."            MAKE  SELECTION." CR CR CR CR
 2  ."         Ø. ABORT TO FORTH   " CR
 3  ."         1. -MUSIC           " CR

1Ø  ."         8. SCR CLR & TITLE  " CR
11  ."         9. ABOUT THIS DISK  " CR  CR CR
12  ." ENTER <DIR> TO RETURN-DON'T FORGET" CR
13  ."      PRESS SPACE FOR PAGE TWO "  CR CR
14  ."           ENTER SELECTION"
15

SCR #31
 Ø  ( MENU SELECT 2 )   DECIMAL
 1  : DIR2  32 LOAD 28 2Ø AT KEY 48 - CASE
 2  -16 OF     DIR                 ENDOF
 3    Ø OF     ABORT               ENDOF
 4    1 OF 7Ø LOAD                 ENDOF

11    8 OF 77 LOAD                 ENDOF
12    9 OF 74 LOAD                 ENDOF
13    ENDCASE ;
14  ' DIR2 POINTER !
15

SCR #32
 Ø  ( MENU SELECTION EGR 7/4/88) HOME
 1  ."            MAKE  SELECTION." CR CR CR
 2  ."         Ø. ABORT TO FORTH   " CR
 3  ."         1. TESTING FORTH    " CR

1Ø  ."         8. ABOUT THE MUSIC  " CR
11  ."         9. ALTER THIS MENU  " CR  CR CR
12  ." ENTER <DIR> TO RETURN-DON'T FORGET" CR
13  ."     PRESS SPACE FOR PAGE ONE" CR CR
14  11 2Ø AT ." ENTER SELECTION "
15
```

```
SCR #63
 Ø  \ DEFER / IS     from Lutz Winkler  12/12/88
 1    FORGET IT  : IT ; \ Invented by M Jaggerman
 2    CLS 1Ø 1Ø AT ." DEFER / IS are loaded" CR CR
 3  ." Enables use of a word not yet defined " CR CR
 4  : DEFER <BUILDS ' CFA , DOES> @@ EXECUTE ;
 5  : IS    CFA [COMPILE]  ' ! ;
 6  \ Usage:  DEFER <anyword>, you may now proceed
 7  \ to use <anyword> in anyway you wish then
 8  \ subsequently define ANYTHING to do something,
 9  \ then write  ' ANYTHING IS <anyword>, as folows
1Ø    DEFER JUNKWORD     \ postpone defining JUNKWORD
11  : PRTMSG JUNKWORD ; \ JUNKWORD used as if defined
12  : TEST CLS 7 12 AT ." JUNKWORD was deferred" ;
13    ' TEST IS JUNKWORD  \ put TEST into JUNKWORD
14    ." enter  JUNKWORD" \ then exercise it
15
```

COPYRIGHTS
by Stan Corbin

I suspect many people who have written programs, have declared the program to be copyrighted without ever having registered them with the Copyright Office. To declare an item as copyright without having registered the same, is a punishable offense. Without registration they have no valid copyright. An unregistered "copyright" has no force in court. That is someone could copy and use the program and would not be liable in a lawsuit. In fact a person who claims a copyright without registering it, may find himself in trouble should he try to sue an "infringer". He could be fined by the copyright office and the so called "infringer" could counter sue for damages.

The information on the following sections are verbatim extractions from the Copyright Law (Public Law 94-553).

Section 106. Exclusive rights in copyrighted works. Subject to sections 107 through 118, the owner of copyright under this title has exclusive rights to do and to authorize any of the following:

(1) to reproduce the copyrighted work in copies or phonorecords;

(2) to prepare derivitave works based upon the copyrighted work;

(3) to distribute copies or phonorecords of the copyrighted work to the public by sale or other transfer of ownership, or by rental, lease, or lending;

(4) in the case of literary, musical, dramatic and choreographicworks, pantomines, and motion pictures and other audiovisual works, to perform the copyrighted work publicly; and

(5) in the case of literary, musical, dramatic, and choreographic works, pantomines, and pictorial, graphic, or sculptural works, including the individual images of a motion picture or other audiovisual work, to display the copyrighted work publicly.

Section 107. Limitations on exclusive rights: Fair use.
Notwithstanding the provisions of section 106, the fair use of a copywrighted work, including such use by reproduction in copies or phonorecords or by any other means specified by that section, for purposes such as critcism, comment, news reporting, teaching(including multiple copies for classroom use), scholarship, or research, is not an infringement of copyright. In determining whether the use made of a work in any particular case is a fair use the factors to be considered shall include--

(1) the purpose and character of the use, including whether such use is of a commercial nature or is for nonprofit educational purposes;

(2) the nature of the copyrighted work;

(3) the amount and substantiality of the portion used in relation to the copyrighted work as a whole; and

(4) the effect of the use upon the potential market for or value of the copyrighted work.

Section 117. Scope of exclusive rights: Use in conjunction with computers and similar information systems.
Notwithstanding the provisions of sections 106 through 116 and 118, this title does not afford to the owner of copyright in a work any greater or lesser rights with respect to the use of the work in conjunction with automatic systems capable of storing, processing, retrieving, or transferring information, or in conjunction with any similar device, machine, or process, than those afforded to works under the law, whether title 17 or the common law or statutes of a State, in effect on December 31, 1977, as held applicable and construed by a court in an action brought under this title.

It is seen that there is some access to the copying of copyright material and under strict circumstances it may be done. Sections 107 through 118 spell out the limitations of the copyright.
I would suggest anyone intending to copyright anything should investigate the copyright laws, and follow their proceedures.

Copies of the copyright laws ($2.25) can be obtained from the GOVERNMENT PRINTING OFFICE or one of its' branches. The Los Angeles branch is located in the Arco Plaza on Level "C", 505 S. Flower, Los Angeles, Ca 90071. Phone 213 894 5841.

Some of the information for this article was obtained from the book AUTHOR LAW STRATEGIES, by Brad Bunnin Peter Beren, copywight 1983, available in the Westminster Library.

Continued from Page 4

The first line is set to the left margin but subsequent lines are indented. It could be called the opposite of normal paragraph indentation.

To do this, you must alter the left margin (.LM) and indentation (.IN) settings. For this column, the initial formatter settings were:

```
.FI;AD;LM 1;RM 40;IN +0
```

This tells the formatter to FIll, ADjust, set the Left Margin at 1, the Right Margin at 40 and NOT to INdent paragraphs.

When I wanted to start the hanging indent, I added this command:

```
.LM +3;IN -3
```

The "+3" after ".LM" tells the Formatter to move the left margin three spaces to the RIGHT. I could have said ".LM 4" but by using the plus sign I don't need to know the previous margin setting. In the same manner, the "-3" INdent tells the formatter to set paragraph indentation three spaces to the LEFT of the left margin.

When I wanted to revert to the original settings, I used this:

```
.LM -3;IN +0
```

The -3 after <.LM> moves the left margin three spaces to the LEFT or back where it was before the <.LM +3> command. The <.IN +0> command cancels the hanging indent (i.e., it tells the Formatter to stop indenting). The + or - in the INdent command is releative to the current left margin.

Notice that I combined a number of formatter commands on the same line. They are separated with semi-colons. This works for most (but not all) formatter commands.

Enjoy.

======================================

UGOC
BULLETIN
BOARD
300/1200/2400 BAUD
(714) 751-4332