

LONG ISLAND
99ER USERS GROUP

VOL #12 NO.07

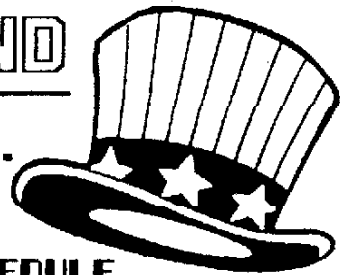
JULY, 1993

\$2.00

NO MEETING JUL (TONY'S fixing up the new house's bosement)



LONG ISLAND SOUND



EDITOR: FRANCIS J. BUBENIK JR.

PAGE 1.....TABLE OF CONTENTS.

PAGE 2.....1993 COMPUTER FAIR SCHEDULE.
COMPILED BY FRANK J. BUBENIK JR.

PAGE 3.....XB MISCELLANEOUS #25.
By EARL RAGUSE / UGOC U.G.

PAGE 4.....XB MISCELLANEOUS #25 (CONTINUED).
COMPLETE "LISTMAN" PROGRAM.

PAGE 5.....TIPS #68 TIGERCUB.
By JIM PETERSON.

PAGE 6.....TIPS #68 TIGERCUB (CONTINUED).
TIGERCUB SOFTWARE.

PAGE 7.....EXTENDED BASIC 3 AD. ASGARD SOFTWARE
By WINFRIED WINKLER.

PAGE 8.....FIRST DRAFT AD. VERSION #2.0.
by ART GIBSON AND ASGARD SOFTWARE.

PAGE 9.....93 COMPRODINE SOFTWARE AD.
By ROGER MERRITT .

PAGE 10.....MICROpendium & HARRISON SOFTWARE AD.
NEW LOGO & MAILER. LI TI 99ERS.

★ ★ HAPPY FOURTH OF JULY ★ ★

ESTABLISHED APRIL-1983

COME TO THE FAIRS

*** 1993 FAIR SCHEDULE ***

Compiled by Frank J. Bubenik Jr. (NL Editor)

JUN 20, 1993 (SUN) BAYSIDE, NY ADRIA HOTEL. 9:30-3:00PM
220-33 NORTHERN BLVD. ADM. \$5.00. 70 TABLES. TSCF.

JUL 18, 1993 (SUN) SECAUCUS, NJ. MEADOWLANDS HILTON.
RTE #3. 9:30-3PM. ADM. \$5.00. TSCF.

JUL 31, 1993 (SAT) HUNTINGTN HILTON. LONG ISLAND SHOW.
MELVILLE, NY EXIT#49-SOUTH RTE#110. 10-3PM 200 TABLES.

AUG 14, 1993 (SAT) FAIRLEIGH DICKINSON UNIV. RTE 4
ON HACKENSACK AVE. HACKENSCK, NJ 10-3PM 500 TABLES.

AUG 28, 1993 (SAT) EDISON NJ. RARITAN CENTER EXP HALL
1200 TABLES. 10-3PM. EXIT#10 NJ TPK. KGP.

SEP 18, 1993 (SAT) WAYNE, NJ. WM PATERSON COLLEGE. 400
TABLES. RTES 46/23. 10-4PM. KGP.

OCT 30/31, 1993 (SAT/SUN) FAIRLEIGH DICKINSON UNIV.
500 TABLES. 10-3PM. RTE 4 TO HACKENSACK AVE. SOUTH. KGP.

TI NOV 12/13, 1993 CHICAGO (FRI/SAT) *11th ANNUAL
INTERNATIONAL FAIRE*. ELK GROVE HOLIDAY INN. HOTEL RES.
708) 437-6010 CODE "IWF". JIM DEARDS NEW CHAIRPERSON.

NOV 13/14, 1993 (SAT/SUN) EDISON, NJ. RARITAN CENTER
1200 TABLES. 10-3PM. EXIT #10 NJ TPK. KGP.

TI NOV 14, 1993 (SUN) MILWAUKEE TI FAIRE. INFO CALL
GENE HITZ (414) 535-0133.

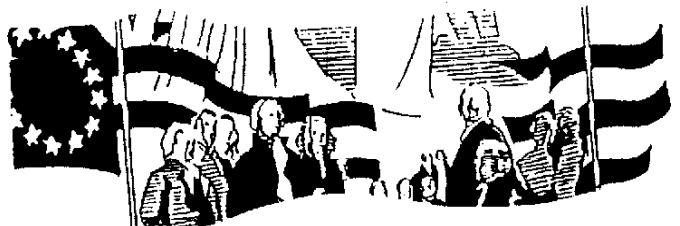
LITI 99ERS NEWSLETTER IS NOT RESPONSIBLE FOR CANCEL-
ATIONS. CALL THE NUMBERS BELOW TO VERIFY TIME AND DATES
BEFORE YOU GO.

* Ken Gordon Productions (KGP) SHOWS COST \$8.00-
\$1.00 discount cards are sent to those people on there
mailing lists. Call (800)631-0062 OR (908)297-2526 for
info. (rev 2/20/93).

* Tri-State Computer Fairs (TSCF) SHOWS COST \$6.00-
\$1.00 discount cards available by mail. Call Robert
Barlow (201) 533-1991 for info. (rev 8/18/93).

Don't miss the bargains on computers, software, IC's,
peripherals, printers, monitors, parts, supplies and
books.

4th of JULY



XB MISCELLANY #25
By Earl Raguse

Well, at long last we are done with the program LISTMAN, as least as far as I am going to take it. I bet I learned as much as you did. Actually I hope you learned as much as I did. On the next page, is a complete listing of the full program.

One thing you may notice, is that there is no provision for deleting a record. You may edit it to a blank, but it is still there. There is a good reason for this. TIF/FW does a better job of this than I could ever do in XB, so why fight it. That's why I chose to use DV80 files. When you go into a file to delete a record, make sure that you delete the whole record, that is a full 6, count em, six lines.

I also did not complete the Search and Sort Modules either. TIW/FW, does a good job of searching for strings using Find String. There is no sense in duplicating that unless you wish to do it for the practice. If so I will be glad to help. If you put your records in order in the first place, you won't be sorting them either. When you add records, do it with TIW/FW, and insert them in the right place.

Even without sorting and searching, you should now be able to write programs that 1. Enter data, 2. Display data, 3. Edit data, 4. Save and Retrieve data from disk files, and 5. Print your data to a printer. That makes you pretty much an expert. You have also learned some programming techniques that only a few high level programmers use. I probably also did some stupid things that I will discover later. If you don't make mistakes, you do not learn anything, so do not fear, make all the mistakes you want, I won't charge you.

If I get enough response from this effort, good or bad, I intend to continue with graphics, I will actually make a LISTMAN subprogram which will have a LISTMAN chasing after lists and eating them if he catches them. If you want me to continue, you must somehow tell me. Write, phone, send Faxes, just tell me, I don't care, but I will not continue unless I know I am not boring you.

Even later, I may do a sort routine, using an assembly sorting LINK.

```

100 ! SAVE DSK1.LISTMAN
110 ! By Earl Raguse
      UGOC 9/92
120 CALL SET(5,16)!
      clear screen,
      set blue color
      white letters
130 F$="A File Has Been Load
ed" :: M$="Restrict Your Sel
ections 1-9" :: P$="PIO"
      :: DIR$="FWD" :: DIM REC$(2
00,6):: DIR=1
135 CALL CLPUT("This is LIST
MAN",4):: CALL PUT("A Produc
t of the",6):: CALL PUT
("UGOC",8):: CALL PUT("Enter
1st letter of a Prompt",12)
138 CALL PUT("Or just follow
instructions ",14):: CALL Q
UIT
140 CALL MENU :: IF N THEN C
ALL PUT(F$,24)
150 CALL GO("123456789",K)::
IF K=0 THEN CALL PUT(M$,23)
:: CALL PAK :: GOTO 140
180 ON K GOTO 1100,1200,1300
,1400,1500,1600,1700,1800,19
00
1100 CALL CLPUT("Data Entry
Section",2)
1110 DISPLAY AT(4,1):"You ma
y have up to 5 lines per re
cord. Use arrow keys, and th
e ENTER key to accept."
1120 IF N THEN 1150 ELSE DIS
PLAY AT(10,1):"If you intend
to append records to an
existing file,you must firs
t load that file."
1130 CALL PUT("If This is an
APPEND file, ",16):: CALL
PUT("Press A, else press any
key ",18)
1140 CALL GKEY(Q,24):: IF Q=
ASC("A")THEN APP=1 :: GOTO 1
400 :: ELSE APP=0
1150 CALL PUT("Data Entry Se
ction",2):: DISPLAY AT(8,1):
"You will over write existin
g records, If that is not wha
t"
1155 DISPLAY AT(10,1):"you w
ant, press A for Addend": :
Press A, Else Press Any Key"
:: CALL GKEY(Q,24):: IF Q=AS
C("A")THEN APP=1 ELSE N=0
1160 N=N+1 :: CALL CLS(7,12)
:: DISPLAY AT(8,7):"Record N
umber";N :: FOR I=1 TO 5 ::

```

```

DISPLAY AT(I+9,1):I :: NEXT
I :: CALL KEY(5,K,S)
1180 CALL KEY(5,K,S):: FOR I
=1 TO 5 :: ACCEPT AT(I+9,3)S
12E(-24):REC$(N,I) :: NEXT I
1190 CALL PUT("Edit More Q
uit",24):: CALL GO("EMQ",Q):
: IF Q=0 THEN 1190 ELSE ON Q
GOTO 1180,1160,140
1200 CALL CLPUT("Data Displa
y Section",2)
1205 IF N<1 THEN CALL PUT("Y
ou Have No Data To Display",
12):: GOTO 1290
1210 CALL CLPUT("If you know
the record",8):: CALL PUT("
number, Enter that numbe
r",10):: CALL PUT("Else Ente
r 00",12):: ACCEPT AT(14,13)
:J
1215 IF J<>0 THEN 1280 ELSE
J=1 :: PAR=0
1220 CALL CLS(6,20):: GOSUB
2100 :: GOTO 1230 ! display
record
1230 CALL PUT("Press First L
etter To",23):: CALL PUT("Ne
xt Last All Quit",24)
1240 CALL GO("MLAQ",Q):: IF
Q=0 THEN 1240 ELSE ON Q GOTO
1250,1260,1275,140
1250 DIR=1 :: GOTO 1265
1260 DIR=-1 :: GOTO 1265
1265 J=J+DIR :: IF J<1 OR J>
N THEN 1270 ELSE IF PAR THEN
1280 ELSE 1220
1270 CALL CLS(22,23):: CALL
PUT("That's All",22):: CALL
PAK :: GOTO 140
1275 PAR=1 ! print all recor
ds
1280 IF J>N THEN 1270 :: CAL
L CLS(6,20):: GOSUB 2100 ::
IF PAR THEN CALL WAIT(400) :
: GOTO 1265 ELSE CALL PAK ::
GOTO 1210
1290 CALL PAK :: GOTO 140
1300 CALL CLPUT("Save Data S
ection",2)
1310 IF N<1 THEN CALL PUT("Y
ou Have No Data To Save",12)
:: GOTO 1390
1320 CALL SAVIT(FIL$,REC$(,
),N,APP)
1390 CALL PAK :: GOTO 140
1400 CALL CLPUT("Load Data S
ection",2)
1410 CALL GETIT(FIL$,REC$(,
),N)
1490 CALL PAK :: GOTO 140

```

continued next page



LISTMAN Listing (Cont)

```

1500 CALL CLPUT("Data Editing Section",2)
1505 IF N<1 THEN CALL PUT("You Have No Data to Edit",12)
:: GOTO 1590
1510 CALL CLS(5,20):: CALL PUT("If you know the record",8):: CALL PUT("number, Enter that number",10):: CALL PUT("Else Enter 00",12)
1515 ACCEPT AT(14,13):: J := 1
IF J<>0 THEN 1575 ELSE J=1 :: CALL CLS(5,20)
1520 DISPLAY AT(7,10):"Record";J :: GOSUB 2100
1530 CALL CLS(22,22):: CALL PUT("Press First Letter To",23):: CALL PUT("Edit Next Last Quit",24)
1540 CALL GO("ENLQ",Q):: IF Q=0 THEN 1540 ELSE ON Q GOTO 1580,1550,1560,140
1550 DIR=1 :: GOTO 1565
1560 DIR=-1 :: GOTO 1565
1565 J=J+DIR :: IF J<1 OR J>N THEN 1570 ELSE IF J>N THEN 140 ELSE 1520
1570 CALL CLS(22,24):: CALL PUT("That's All",22):: GOTO 1520
1575 CALL CLEAR :: IF J>N THEN 1570 :: DISPLAY AT(7,10):"Record";J :: GOSUB 2100
1580 FOR I=1 TO 5 :: CALL KEY(Y(5,K,S)):: ACCEPT AT(I+9,3)SIZE(-24):REC$(J,I):: NEXT I :: CALL PUT("More Edit Quit",23)
1585 CALL GO("MEQ",Q):: IF Q=0 THEN 1585 :: ON Q GOTO 1510,1580,140
1590 CALL PAK :: GOTO 140
1600 CALL CLPUT("Print Labels Section",2)
1610 IF N<1 THEN CALL PUT("You Have No Data to Print",12)
:: GOTO 1690
1620 M=4 :: CALL OPENPRNTRIP$(M):: CALL EMPHASIZE(M):: J,DIR=1 :: PAR=0
1625 DISPLAY AT(7,10):"Record":J
1630 CALL CLS(5,5):: CALL CLS(24,24):: GOSUB 2100 <: IF PAR THEN 1680
1640 CALL PUT("Next Last Print All Quit",23):: PAR=0
1645 CALL GO("NLPAQ",Q):: IF Q=0 THEN 1645 ELSE ON Q GOTO 1650,1655,1680,1670,1685
1650 DIR=1 :: GOTO 1660
1655 DIR=-1 :: GOTO 1660
1660 J=J+DIR :: IF J<1 OR J>N THEN 1685 ELSE 1625
1670 PAR=1 :: CALL PUT("To Halt, Hold H Key",5): print a 1) records
1680 PRINT #N:CHR$(10):: FOR

```

```

K=1 TO 5 :: PRINT #N:REC$(J,K):: NEXT K :: CALL KEY(3,K,S):: IF K=ASC("H")THEN 1640 ELSE 1660
1685 CLOSE #N :: CALL CLS(22,23):: CALL PUT("That's All",20)
1690 CALL PAK :: GOTO 140
1700 CALL CLPUT("Data Sort Section",2)
1710 CALL PUT("Haven't Done That Yet",12)
1790 CALL PAK :: GOTO 140
1800 CALL CLPUT("Data Searching Section",2)
1810 CALL PUT("Haven't Done That Either",12)
1890 CALL PAK :: GOTO 140
1900 CALL CLPUT("Help Section",2)
1910 CALL PUT("Read DV80 File Called",12):: CALL PUT("LISTHELP",14)
1920 CALL PUT("You May Have To Write That",16)
1930 CALL PUT("Do You Want To Head Now? Y/N",18):: CALL GKEY(Q,24):: IF Q=ASC("Y") THEN RUN "DSK1.FILEREAD" ELSE 1990
1990 CALL PAK :: GOTO 140
2000 !
2010 ! Subroutine Area
2020 !
2100 DISPLAY AT(7,10):"Record";J :: FOR K=1 TO 5 :: DISPLAY AT(K+9,1):K:REC$(J,K) :: NEXT K :: RETURN
2970 !
2980 ! Subprogram area
2990 !
3500 SUB MENU :: CALL CLPUT("LISTMAN",1):: CALL PUT("Press Number to Select",3)
3510 DISPLAY AT(05,9):"1. Enter Data"
3515 DISPLAY AT(07,9):"2. Display Data"
3520 DISPLAY AT(09,9):"3. Save Data"
3525 DISPLAY AT(11,9):"4. Load Data"
3530 DISPLAY AT(13,9):"5. Edit Data"
3535 DISPLAY AT(15,9):"6. Print Labels"
3540 DISPLAY AT(17,9):"7. Sort Data"
3545 DISPLAY AT(19,9):"8. Search Data"
3550 DISPLAY AT(21,9):"9. Get Help"
3560 SUBEND
4000 SUB GKEY(Q,ROW)
4010 CALL KEY(3,K,S):: IF S<1 THEN 4010 :: DISPLAY AT(ROW,1)SIZE(30):" You Selected ";CHR$(K):: Q=K
4020 SUBEND
4100 SUB SAVIT(FIL$,VAR$(,),

```

```

N,APP)
4110 CALL PUT("What Drive?",12):: CALL GKEY(Q,24):: IF (Q-48)<1 THEN 4110 ELSE D8=CHR(Q): CALL PUT("What File Name",14)
4120 DISPLAY AT(16,10):FIL$:4130 ACCEPT AT(16,10)SIZE(-10):FIL$ :: OPEN #3:"DSK1.D8$":FIL$,OUTPUT,DISPLAY,VARIABLE 80
4140 !PRINT #3:N
4150 FOR J=1 TO N :: PRINT #3:CHR$(32):: FOR K=1 TO 5 :: DISPLAY AT(12,6):"Saving Record";J :: PRINT #3:VAR$(J,K) :: NEXT K :: NEXT J :: CLOSE #3 :: N=J-1
4160 DISPLAY AT(12,1)ERASE ALL:" I Saved";N:"Records" :: SUBEND
4200 SUB GETIT(FIL$,VAR$(,),N)
4210 CALL PUT("Enter Drive",12):: CALL GKEY(Q,24):: IF (Q-48)<1 THEN 4210 ELSE C%=CHR$(Q):: ON ERROR 4215
4215 CALL PUT("Enter File Name",12):: CALL PUT("if you can't remember ",20):: CALL PUT("Say 00 for Disk Director Y",22)
4220 DISPLAY AT(16,10):FIL$:ACCEPT AT(16,10)SIZE(-10):FIL$ :: IF FIL$="00" THEN 4215 ALL CAT :: CALL CLEAR :: GOT 0 4215
4225 OPEN #3:"DSK1.D8$":FIL$,INPUT :: DISPLAY AT(10,8):"Reading Record":ON ERROR STOP
4230 FOR J=1 TO 200 :: DISPLAY AT(10,22):J
4235 IF EOF(3)THEN 4250 :: LINPUT #3:A$
4237 FOR K=1 TO 5
4240 IF EOF(3)THEN 4250 :: LINPUT #3:VAR$(J,K):: DISPLAY AT(9+K,2):VAR$(J,K)! Optional display
4245 NEXT K :: NEXT J
4250 CLOSE #3 :: CALL CLPUT("I Read "STR$(J-1)&" Record s",12):: N=J-1 :: SUBEND
4300 SUB CLS(R1,R2):: FOR R=R1 TO R2 :: DISPLAY AT(R,1)SIZE(30):: NEXT R :: SUBEND
4700 SUB WAIT(T):: FOR WAIT=1 TO T :: NEXT WAIT :: SUBEND
4800 SUB CLPUT(A$,L)
4810 T=15-LEN(A$)/2 :: DISPLAY AT(L,T)ERASE ALL:A$
4820 SUBEND
4900 SUB PUT(A$,L)
4910 T=15-LEN(A$)/2 :: DISPLAY AT(L,1)SIZE(30):: DISPLAY AT(L,T):A$
4920 SUBEND
5500 SUB AGAIN :: DISPLAY AT(24,1):"Again? Press A, Else

```

```

Any Key"
5510 CALL KEY(3,K,S):: IF S<1 THEN 5510 ELSE IF K=ASC("A")THEN RUN "DSK1.D1R"
5520 SUBEND
5900 SUB GO(A$,K):: CALL GKEY(Y(Q,24)): K=POS(A$,CHR$(Q),1):: SUBEND
6100 SUB PAK
6110 DISPLAY AT(24,3)SIZE(30):"Press Any Key to Proceed"
6120 CALL KEY(3,K,S):: IF S<1 THEN 6110
6130 SUBEND
6150 SUB QUIT
6160 DISPLAY AT(23,4):" Press Any Key to Proceed " 8 ut Q Quits, and B Breaks"
6170 CALL KEY(0,K,S):: IF S<1 THEN 6160 ELSE K=K AND 95 :: IF CHR$(K)="Q" THEN RUN "DSK1.D1R" ELSE IF CHR$(K)="B" THEN STOP
6180 SUBEND
6200 SUB SET(S,C):: CALL CLEAR :: CALL SCREEN(S):: CALL CHARSEP :: FOR I=0 TO 14 :: CALL COLOR(I,C,1):: NEXT I :: SUBEND
6220 SUB PAUSE :: DISPLAY AT(24,1):"SPACE Pauses, But Q Quits"
6230 CALL KEY(3,K,S):: IF S<1 THEN 6250 ELSE IF CHR$(K)="Q" THEN 130 ELSE IF K=32 THEN 6230
6240 DISPLAY AT(24,1)SIZE(30)
6250 SUBEND
6300 SUB CAT
6305 ! Needs GKEY, PAK & CLS
6310 ! E Raguse 10/90
6320 DISPLAY AT(10,1)ERASE ALL:" Enter Drive# to Catalog" :: CALL GKEY(Q,24):: D%=CHR$(Q):: CALL CLS(10,11)
6330 OPEN #1:"DSK1.D8$":RELATIVE,INTERNAL,INPUT
6340 INPUT #1:A$,L,L ! read and toss
6350 X,Z=0 :: FOR F=1 TO 127 :: INPUT #1:A$,T,J,K :: IF A$="" THEN F=17
6360 Y=2-14*(Z=1)
6365 IF ABS(T)=2 THEN DISPLAY AT(X+3,Y)SIZE(14):A$ ELSE X=X-1 :: GOTO 6370
6370 IF X<18 THEN X=X+1 ELSE X=0 :: Z=1-Z
6380 NEXT F :: CLOSE #1 :: CALL PAK :: SUBEND
6400 SUB OPENPRNTR(PRNTR$,N) :: DISPLAY AT(12,1)ERASE ALL:" TURN ON YOUR PRINTER"
6410 OPEN #N:PRNTR$,OUTPUT,VARIABLE 136 :: PRINT #N:CHR$(27)&CHR$(64):: DISPLAY AT(12,1)SIZE(30):: SUBEND
6560 SUB EMPHASIZE(N):: PRINT #N:CHR$(27):"E" :: SUBEND

```

TIPS FROM THE TIGERCUB

No. 68

Tigercub Software
156 Collingwood Ave.
Columbus, OH 43213

My three Nuts & Bolts disks, each containing 100 or more subprograms, have been reduced to \$5.00 each. I am out of printed documentation so it will be supplied on disk.

My TI-PD library now has almost 600 disks of fairware (by author's permission only) and public domain, all arranged by category and as full as possible, provided with loaders by full program name rather than filename, Basic programs converted to XBasic, etc. The price is just \$1.50 per disk(!), post paid if at least eight are ordered. TI-PD catalog \$5 and the latest supplement is available for \$1 which is deductible from the first order.

When I have finished reading Barry Traver's column in Computer Monthly, I like to take a look at whatever Dr. Michael Ecker is up to in his "Recreational Computing" column, although much of his math is beyond me and I can't always translate his GW Basic into TI Basic.

In the February issue, he had a routine to play Fibonacci modular music. This is the TI version; it is not very musical, but the notes are in the chromatic scale.

```
100 A=0 :: B=1 :: M=51
110 C=A+B :: C=C-M:INT(C/M):
:: CALL SOUND(-100,110*2^(C/12),5):: A=B :: B=C :: GOTO 110
```

Dr. Ecker also had a challenge to swap two numbers without using a third vari-

able or the SWAP command - which TI Basic doesn't have anyway. The practical way, of course, is to use the 3rd variable, T=A :: A=B :: B=T, but just for the fun of it, if we are dealing with one-digit numbers -

```
100 A=1 :: B=2 :: A=A+B/10 :: B=INT(A):: A=(A-INT(A))*10 :: PRINT A;B
```

But suppose we are dealing with numbers of any length - we can still do it with a one-liner, or a two-liner if we want to input the numbers from the keyboard -

```
100 INPUT A :: INPUT B
110 B=B/10^(LEN(STR$(B))):
A=A+B :: B=INT(A): A=A-INT(A):
A=A*10^(LEN(STR$(A))-1)
:: PRINT A;B :: GOTO 110
```

So you got smart and tried a negative number or a decimal? OK, how about this -

```
100 INPUT A$ :: INPUT B$
110 A$=A$+" "&B$ :: B$=SEG$(A$,1,POS(A$," ",1)-1):: A$=SEG$(A$,POS(A$," ",1)+1,255):
:: PRINT A$;" ";B$ :: GOTO 110
```

And another challenge was to alternately assign X the value of A and B, without using IF...THEN or any outside help. That seems to require a two-liner -

```
100 A,X=77 :: B=132
110 X=ABS(X=A)*B+ABS(X=B)*A
:: PRINT X :: GOTO 110
```

The only honest way to compute interest on a loan is on the unpaid balance, although the banks and finance companies have devised more complicated and profitable ways. If you want to make an honest loan, here is how to do it -

```
100 DISPLAY AT(3,1)ERASE ALL
:"SIMPLE INTEREST CALCULATOR
":": "For interest to be cal-
```

culated monthly on unpaid balance."

```
110 DISPLAY AT(9,1):"Printer ? PID" :: ACCEPT AT(9,10)SIZE(-20):P$
```

```
120 DISPLAY AT(11,1):"Amount loaned? $" :: ACCEPT AT(11,17)VALIDATE(NUMERIC):A
```

```
130 DISPLAY AT(13,1):"Interest rate? %" :: ACCEPT AT(13,16)SIZE(4)VALIDATE(NUMERIC):X
```

```
140 IF X<1 THEN DISPLAY AT(12,1):"Enter as a percentage" :: GOTO 130
```

```
150 DISPLAY AT(15,1):"Monthly payments of $" :: ACCEPT AT(15,22)VALIDATE(NUMERIC):P
```

```
160 DISPLAY AT(17,1):"Beginning in month (1-12) of year"
```

```
170 ACCEPT AT(17,27)VALIDATE(DIGIT):M :: ACCEPT AT(18,9)VALIDATE(DIGIT):Y
```

```
180 DATA JAN,FEB,MAR,APR,MAY,JUN,JUL,AUG,SEP,OCT,NOV,DEC
```

```
190 X=X/100 :: DIM M$(12):: FOR J=1 TO 12 :: READ M$(J):
```

```
200 OPEN #1:P$,VARIABLE 254 :: PRINT #1:CHR$(27)&"E"&CHR$(27)&"G"&CHR$(27)&"M"&CHR$(27)&"N";
```

```
210 PRINT #1:"$";STR$(A);" FINANCED AT ";STR$(X*100);"% WITH MONTHLY PAYMENTS OF $";
```

```
STR$(P);" BEGINNING ";M$(M);Y:""
```

```
220 I=A*X/12 :: TI=TI+I :: A=A+I-P
```

```
230 PRINT #1:M$(M);Y;" PAYMENT $";STR$(P);" OF ";
```

```
240 PRINT #1,USING "$###.##":I;:: PRINT #1:" INTEREST AMOUNT $";
```

```
250 PRINT #1,USING "$###.##":P-I;:: PRINT #1:" PRINCIPAL - BALANCE OF $";
```

```
260 PRINT #1,USING "$###.##":A
```

```
270 M=M+1 :: IF M=13 THEN M=1 :: Y=Y+1
```

```
280 IF A>P THEN 220
290 PRINT #1,USING "FINAL PAYMENT $###.##":A :: PRINT #1,USING "TOTAL INTEREST PAYED $###.##":TI
```

Thanks to Bruce Harrison, here is a neat subprogram to

sort strings into sequence as they are entered -

```
100 CALL CLEAR :: DIM W$(100)
```

```
110 FOR J=1 TO N :: W$(J)=" " :: NEXT J :: INPUT "N=? ":N
```

```
120 INPUT I$ :: IF I$="" THEN N 130 ELSE CALL INSORT(W$(1),I$,N):: GOTO 120
```

```
130 FOR J=1 TO N :: PRINT W$(J):: NEXT J :: GOTO 110
```

```
30020 SUB INSORT(W$(1),I$,N): FOR T=1 TO N : IF I$>W$(T) THEN 30030 ELSE 30040
```

```
30030 NEXT T :: GOTO 30050
30040 FOR J=N TO T STEP -1 : W$(J+1)=W$(J) : NEXT J
30050 W$(T)=I$ :: N=N+1 :: RETURN
```

In the test routine in lines 100-130, give N the value of 0, input some words and then just press enter.

To start a new array, use FOR J=1 TO N :: W\$(J)=" :: NEXT J, then reset N to 0. If you want to sort in reverse sequence, change the > to <. If you need to sort numbers, delete all the \$, change the " " in line 120 to 0, and input a 0 when you are when finished inputting.

Someone sent me a program to figure days between dates but it would not count leap dates, so I decided to write one that would.

```
100 DISPLAY AT(2,5)ERASE ALL
:"DAYS BETWEEN DATES:":: including leap year days" :: M$(1)="From" :: M$(2)="To" :: R=13
```

```
110 DATA 31,28,31,30,31,30,31,31,30,31,30,31
```

```
120 DIM L(12):: FOR J=1 TO 12 :: READ L(J):: NEXT J
```

```
130 FOR J=1 TO 2 :: DISPLAY AT(R-1,1):M$(J):"year month day " :: ACCEPT AT(R,6)VALIDATE(DIGIT)SIZE(4):Y(J)
```

```
140 ACCEPT AT(R,17)VALIDATE(DIGIT)SIZE(2):M(J):: IF M(J)<1 OR M(J)>12 THEN 140
```

```
150 ACCEPT AT(R,24)VALIDATE(DIGIT)SIZE(2):D(J):: IF D(J)<1 OR D(J)>L(M(J)) THEN 150
```

```

D15(1)S12(2):D(J):: IF D(J)
<1 OR D(J)>31 THEN 150
160 CALL LEAP(Y(J),X):: L(2)
=L(2) X :: IF D(J)>L(N(J))TH
EN 150
170 L(2)=28 :: R=R+3 :: NEXT
J :: R=13 :: IF Y(1)>Y(2)TH
EN T=Y(1):: Y(1)=Y(2):: Y(2)
=T :: T=M(1):: M(1)=M(2):: M
(2)=T :: T=D(1):: D(1)=D(2)::
D(2)=T
180 IF Y(1)=Y(2)AND M(1)>M(2)
THEN T=M(1):: M(1)=M(2):: M
(2)=T :: T=D(1):: D(1)=D(2)::
D(2)=T
190 L(2)=28 :: IF Y(2)>Y(1)T
HEN 220
200 IF M(1)=M(2)THEN B=ABS(D
(2)-D(1)):: GOTO 260
210 CALL LEAP(Y(1),X):: FOR
J=M(1)+1 TO M(2)-1 :: B=B+L(
J)+X*(M(1)=2):: NEXT J :: B=
B+L(M(1))+X*(M(1)=2)-D(1)+D(
2):: GOTO 260
220 CALL LEAP(Y(1),X):: B=L(
M(1))-D(1)+X*(M(1)=2)
230 FOR J=M(1)+1 TO 12 :: B=
B+L(J)+X*(J=2):: NEXT J
240 FOR J=Y(1)+1 TO Y(2)-1 :
: CALL LEAP(J,X):: B=B+365-X
:: NEXT J
250 FOR J=1 TO M(2)-1 :: CAL
L LEAP(Y(2),X):: B=B+L(J)+X*(
J=2):: NEXT J :: B=B+D(2)
260 DISPLAY AT(20,1):B;"days
between" :: B=0 :: GOTO 130
270 SUB LEAP(Y,X):: X=(Y/400
=INT(Y/400)):: IF X=-1 THEN
SUBEXIT ELSE X=(Y/4=INT(Y/4)
):: IF X=0 THEN SUBEXIT ELSE
X=(Y/100<)>INT(Y/100)
280 SUBEND

```

A leap year is a year that is evenly divisible by 4 unless it is evenly divisible by 100 but not evenly divisible by 400. The subprogram in lines 270-280 will give X a value of -1 if Y is a leap year.

Gene Hitz of Arcade Action Software reports another undocumented feature of TI Extended Basic. The manual says that you can only enter a subprogram by a CALL and only leave it by a SUBEXIT or SUBEND, but the manual is

wrong. You can GOSUB to a subroutine within a subprogram, providing it does not contain a SUBEXIT, and return; and you can GOSUB from within a subprogram to a subroutine in the main program, and return. In this way, you can transfer variables in and out of a subprogram without putting them in a parameter list. See for yourself -

```

100 CALL CLEAR
110 INPUT M$ :: CALL SUB(M$)
:: PRINT M$ :: GOSUB 140 ::
PRINT "M$ IS";X;"CHARACTERS
LONG" :: GOTO 110
120 M$="SEE WHAT I TOLD YOU?"
:: RETURN
130 SUB SUB(M$):: GOSUB 120
:: GOSUB 140 :: SUBEXIT
140 X=LEN(M$):: RETURN
150 SUBEND

```

If you are among the lonely few who have purchased my TI-PD disks, you will know that most of them load from a menu by full program name, not those abbreviated file-names. Those menus are prepared quickly and easily by my Catwriter program which was published in Tips #47 and in MICROpendium and is available on TI-PD 1105.2.

I was asked if there was a way to dump those full program names to the printer. There is, but it requires a big program - like this -

```

1 OPEN #1:"DSK2.TI-PD/CAT",A
PPEND
2 DISPLAY AT(12,1)ERASE ALL:
"TI-PD# ?" :: ACCEPT AT(12,1
0):N
14 FOR J=1 TO X-1 :: READ X$
:: PRINT #1:X$;TAB(30);N ::
NEXT J :: CLOSE #1 :: STOP
17 REM

```

Save that on an empty disk by SAVE DSK2.C,MERGE. Put your TI-PD disk in drive 1, boot its LOAD program, break it with FCTN 4 and enter MERGE DSK2.C, then RUN. Put

in the next TI-PD disk and do the same. You will have a D/V80 file of all the programs, followed by their TI-PD disk number. Run the file through Sort Experiaent or TI-Sort or whatever, and you can print them out in alphabetical sequence.

If you have only one drive just change that DSK2. to DSK1. and swap disks after breaking the LOAD program.

Of course, this won't work with fairware disks which have the author's own loader or some other disks which do not have my Catwriter load for one reason or another. You'll have to type those into the file.

Another user asked me if there was anyway to key in the ASCII above 127 into TI-Writer's Editor. Many of those ASCII can be entered from the keyboard by using the CTRL and FCTN keys - try this -

```

100 INPUT M$ :: PRINT ASC(M$
): GOTO 100

```

- but the Editor has been programmed to refuse them because so many of those FCTN and CTRL combinations are used as edit commands.

I had a bright idea - I thought. I wrote a little program to create 127 files, named 128 through 255, each containing just the ASCII of the same number. Now, I thought, when I want to put in such an ASCII I will just LF that file into the next line and CTR 2 to pop it in to place. But the Editor refused to even load a file that began with an ASCII above 127!

I'll fool you, I thought. I created those files again, but with an asterisk before the high ASCII. Now they loaded alright - but each ASCII above 127 became an ASCII 128 numbers lower! It is too bad that the Editor does not have a command to

add 127 to an ASCII, just as CTRL U subtracts 64, but if you want those graphics characters in your text you will just have to transliterate them and print through the Foreatter.

Folks take it for granted that my Nuts & Bolts disks are only useful for programmers, but they contain many routines so simple to use that anyone can use them to dress up their favorite program. For instance -

```

20083 SUB TITLE(S,T$):: CALL
SCREEN(S):: L=LEN(T$):: CAL
L MAGNIFY(2)
20084 FOR J=1 TO L :: CALL S
PRITE(#J,ASC(SEG$(T$,J,1)),J
+1-(J+1=S)+(J+1=S+13)+(J+14)
#13,J*(170/L),10+J*(200/L)):
: NEXT J
20085 SUBEND

```

Key that in and save it by SAVE DSK1.TITLE,MERGE. Load your favorite program. Enter MERGE DSK1.TITLE. Make sure your program does not have a line 1 or 2 - if so, RES it. Type in -

```

1 CALL CLEAR :: CALL TITLE(5
,"MY PROGRAM")
2 FOR D=1 TO 1000 :: NEXT D
:: CALL DELSPRITE(ALL)

```

And try it. Instead of "MY PROGRAM", put the name of your program. Instead of 5, put the number of whatever screen color you would like, from 2 to 16 - check your Basic manual. Change 1000 to whatever delay you want - if you have selected a screen color that will leave text legible, use -

```

2 DISPLAY AT(24,1):"PRESS AN
Y KEY" :: DISPLAY AT(24,1):"
press any key" :: CALL KEY(0
,K,S):: IF S=0 THEN 2 ELSE C
ALL DELSPRITE(ALL)

```

You might also need a CALL SCREEN(8) to restore normal screen color.

Oops! Memory full! - Jim P

END

ANNOUNCEMENT

Subject: *First Draft 2.0 Now Shipping!*

Contact: Asgard Software
1423 Flagship Dr.
Woodbridge, VA 22192
(716)778-9104 (10AM-6PM EST, Mon-Fri)
(703)491-1267 (7PM-10PM EST, Mon-Sat)

Art Gilman and Asgard Software are pleased to announce the release of *First Draft* version 2.0.

This comprehensive update offers many enhancements to the operation and capabilities of an already remarkable word processor. The only popular word processor currently available that is not a re-write of *TI-Writer*, the update expands and improves on its many unique features.

Version 2.0 offers the following:

- **IMPROVED INSTALLATION.** *First Draft* was the first (and so far only) word processor that can be fully customized. When you install *First Draft* it asks you a range of questions. Your answers are then embedded within the program - including screen colors, tabs, default disk drives and much more. Version 2.0 now allows you to install the program even if you only have a single disk drive.
- **IMPROVED DISK ACCESS SPEED.** *First Draft 1.0* was the first (and only) word processor for the 4A that didn't limit your text file to what you can cram into RAM - the only limit was your available disk space. In *First Draft 2.0* this is still true, but disk access is many times faster and more intelligent. If you have a RAM-disk, you most likely won't even lose a keystroke when it goes to disk with version 2.0.
- **NEW KEYBOARD BUFFERING.** Do you type too fast for *TI-Writer* or its clones? If you are frustrated by dropped letters at the end of your sentences, you may be interested to know that *First Draft 2.0* offers advanced buffering of the keyboard. In fact, it is virtually impossible to out-type *First Draft 2.0*!
- **COMPLETELY NEW MANUALS.** While our original manual was pretty good, we took all the comments we received from our customers and completely re-wrote the manual for version 2.0 from scratch. The result is a much more professional, easy to follow guide to using *First Draft*.

In addition to the many major changes in version 2.0, there are also many lesser ones, including faster search and replace, faster line deletion, paragraph reformatting that can be controlled, enhancements to the *Final Copy* formatter, access to all control codes and character graphics in your document, and more.

As with version 1.0, *First Draft 2.0* still offers many unique features unmatched by other word processors, including the ability to:

- Write in parallel columns on the screen in *First Draft*, and format in newsletter-style columns in *Final Copy*
- Include *TI-Artist* instances and *Page Pro 99* pictures in your documents for logos, headlines and much more
- Use built-in print drivers or create your own. Print drivers allow you to take advantage of your printer without putting printer codes in your document - so your documents will print the same on any printer from an old *TI Impact Printer* to a new *HP Laserjet 4*
- Manage disks with a built-in disk manager with support for *DISKU* comments
- Use the program from a "pull-down menu" user interface - no command line
- View your formatted document before printing with the *Screen Preview* function built into the *Final Copy* formatter
- Take advantage of dozens of new formatting commands that let you to control everything about your printer and document with simple to use commands, and still use all of the old *TI-Writer* formatting commands too.
- And of course, much more

First Draft 1.0 was the first all-new word processor for the 80/4A in years, and offered many promising new features found nowhere else. *First Draft 2.0* fulfills the promise of version 1.0 with a mature program that lets you do things you would otherwise have to buy another computer to do.

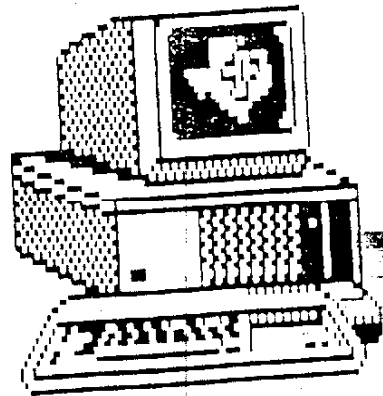
- **IMPROVED 40-COLUMN SUPPORT.** In *First Draft 1.0* users without 80-column cards could only create documents 40 characters wide. To print in 80-columns you have to use the program's *Final Copy* formatter. This limitation is gone - in spades - in version 2.0! Not only do you get an 80-column wide page with a standard 99/4A, but the screen window scrolls over as you type (like *WordPerfect*), and again, you never lose a keystroke! As always, you can use the 80-column display offered by the Geneve and all *TI 80-column* cards.
- **FASTER SPELL-CHECKING.** *First Draft 1.0* was one of the few *TI* word processors with a spelling checker built-in, and the only one currently available. This spelling checker allows you to check a word or the whole document while you are writing it. *First Draft 2.0* is now officially the **FASTEST SPELLING CHECKER** for the 99/4A or the Geneve. This new and improved checker even beats out the previous speed champ - our own *Spell It!* *First Draft 2.0*'s spelling checker will find any word in the included 25,000 word dictionary in a seconds - even from floppy disk.
- **EXTENDED MEMORY SUPPORT.** Even though *First Draft* doesn't have a limit on the size of documents you can create, more memory will make any program faster. *First Draft 2.0* is the first 99/4A word processor designed to take advantage of the new generation of Extended Memory cards for the 99/4A - AMS and AEMS-compatible memory cards. If *First Draft 2.0* sees an AMS compatible card, it will automatically put all of itself into it, and use the remaining space to store text before it is sent to disk. A 128K AMS will store 70K of text in memory, and let you move between the *First Draft* editor and the *Final Copy* formatter at the press of a button - and your document stays in memory in the process. With a 512K AMS or AEMS, you can have over 450K of text in memory; more than *WordPerfect 5.1* lets you have on a 640K PC!
- **MAKE KEYBOARD MACROS.** *First Draft 2.0* is the only word processor for the 4A or the 9640 that lets you define up to of 11 keyboard macros for commonly used text - and load or save those macros to disk.
- **FULL GENEVE COMPATIBILITY.** In contrast to version 1.0, *First Draft* has been modified and verified to work fully on a Geneve with 1.14P of MDOS and 1.04 of the GPL Interpreter. Because we thought this was the case in version 1.0, we are giving all registered owners of *First Draft 1.0* a free upgrade to 2.0. We value our decade-long relationship with our customers more than anything else.

ORDERING INFORMATION

First Draft 2.0 has a suggested retail price of \$39.95. This includes 2 disks and 2 manuals. Until August 1, 1993, registered users of *Spell It!* can purchase the program for \$29.95 (to order, include a photocopy of your program disk). Updates from version 1.0 will be mailed free of charge to registered users.

To order, send a check or money order, plus \$3.00 S&H (overseas Airmail, \$7.00) to:

Asgard Software
1423 Flagship Dr.
Woodbridge, VA 22192



NEW FOR 1993 FROM COMPRODINE

COMPRODINE announces the release of 4 new products for the TI-99 and 9640 users. Software requires TI or 9640, XB or EA, 32K memory and disk drive.

ARTIST CARDSHOP- by Paul Coleman is our newest 4-fold card-making utility. Fast, flexible, easy and friendly, this program will step you through your card creation in just minutes. When done it will print the card(s 0-99), ready to fold, sign and send.

This program utilizes TI-ARTIST compatible Fonts and Instances which means there are hundreds of Fonts and thousands of instances (pictures) to draw from in the TI world. The card may have 2 Fonts on the front and 2 different Fonts on the inside page. You may also use up to 4 Instances on each of the front and inside page. One Instance can be placed on the back page for perhaps a logo.

*FEATURES- Easy step-through menu system
Uses TI-ARTIST fonts and instances
Proportional spacing of letters
Pictures and text on same lines
Pictures may be mirrored
May have a row of pictures across page
Double density double pass printing for good results
25 borders included, plus a blank border
A border maker to make your own borders
Extra- special font and instance utilities
A 20 page info-packed manual with pictures of borders



PRICE- \$20 + \$1.50 postage

WAR ZONE II- by Quinton Tormanen is similar to WAR ZONE but set in the future and each of its 20 levels has different graphics with real trees, fields, rivers, lakes and roads with moving vehicles. High scores save to disk and scoring guide displayed on screen. Greater logic is required of this arcade style game. PRICE \$10 + \$1 postage requires joystick

MINE FIELD- by Quinton Tormanen is a mind teasing logic game. The object is to clear a mine field by exposing only the safe areas. 10 sizes to choose from. You will get either a time if you win or a percentage completed. This will be saved to disk. This is based on an IBM Windows game but has more options and plays faster than a 386-33 machine. Because there is some element of luck, you may choose up to 3 mistakes per game. Winning scores will be saved to disk. PRICE \$10 + \$1 postage

BACKSTEINE SCREENS- by Steve Mehr are 50 new well designed screens for Backsteine users. Each screen has been tested for playability and uniqueness. A new version 1.2 of Backsteine is included with these new screens. An install program updates your version and places the new screens onto your old copy. PRICE \$7 + \$1 postage

We have many new programs coming out in 1993 including games and color printing programs for color printer owners.
To order, mail check or money order to:

Rodger Merritt 1949 Evergreen Ave. Fullerton, CA 92635 (714) 990-4577 12-9pm PT

