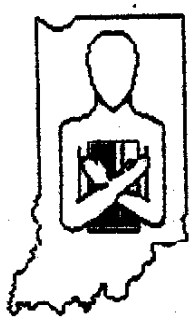


HOOSIER USERS GROUP HOOSIER USERS GROUP HOOSIER USERS GROUP HOOSIER USERS GROUP HOOSIER USERS GROUP
USERS GROUP HOOSIER USERS GROUP HOOSIER USERS GROUP HOOSIER USERS GROUP HOOSIER USERS GROUP HOOSIER
GROUP HOOSIER USERS GROUP HOOSIER USERS
HOOSIER USERS GROUP THE HUGgers HOOSIER USERS GROUP
USERS GROUP HOOSIER People Helping People
GROUP HOOSIER USERS
HOOSIER USERS GROUP
USERS GROUP HOOSIER USERS GROUP HOOSIER USERS GROUP HOOSIER USERS GROUP HOOSIER USERS GROUP HOOSIER
GROUP HOOSIER USERS GROUP HOOSIER USERS GROUP HOOSIER USERS GROUP HOOSIER USERS GROUP HOOSIER USERS

JANUARY, 1986 THE HUGgers NEWSLETTER VOLUME 3, NUMBER 10

THE OFFICER'S CORNER

Another year has come and gone and the future of the computer that refuses to die looks brighter than ever. With the expected arrival of a new computer by Myarc and new products recently released such as Millers Graphics Gram Kracker and Orphan Chronicles; Myarc's Ram Disk and Extended Basic II; and New Horizons Ram Disk brings new life to our old friend.

The winner of the Myarc Ram Disk raffle is John Powell. I'd like to thank all members who participated in our first raffle; and thank Larry Conner, who furnished the Ram Disk and donated half its cost to the users group.

This month's workshops are the Programmers' Corner and the Gram Kracker. G Greg Larson has more information on the Programmers' Corner on page 3 of this New sletter. I'll be giving a demo on the Gram Kracker. It will be a learning experience for me since I haven't used the Gram Kracker to its full potential. One of the workshops will be delayed so everyone will be able to sit in on both.

See you January 12, at 2:00 pm.
J.Steven Sims

SOUTH REGIONAL MEETING

Because of several scheduled activities, I will not be able to have a southside meeting during the month of January. I will speak with some of the members who usually attend the southside meetings to arrange for a February date. The February date will be printed in the Feb. newsletter.

THE NEXT MONTHLY MEETING WILL BE
JANUARY 12, 1986
STARTING AT 2:00 PM AT CREATIVE LOGIC.
SEE YOU THERE!

WORKSHOPS: GRAM KRACKER
PROGRAMMERS' CORNER

'TWAS THE NIGHT BEFORE CHRISTMAS

(A true story)
by Pam Sims

Twas the night before Christmas
and all through the house,
not a creature was stirring,
not even our mouse.
My stocking was hung
by my computer with care,
in hopes that St. UPS
soon would be there.
With visions of Bonz and mice
dancing in their heads...
Snoopy and Bootz,
our dog and our cat,
had just settled in
for their long winter's naps.
When from outside a rumbling we heard,
Snoopy and I sprang from the sofa
to see what had occured.
A skinny young fella.
so tired and sick
had just pulled up
in a brown delivery truck mighty quick.
He searched through his parcels
till he found just the right one,
I was so anxious to get my Gram Kracker,
but I knew I'd have to wait till he was done.
Snoopy was a growling,
and I was a shaking,
both knowing that the skinny young fella
soon would be waiting.
"Sign line three
for your Gram Kracker please"
the skinny young fella said with glee.
I said "Thank you" on this Christmas Eve.
As he turned and walked away
I could hear him say
"Merry Christmas to all,
and, to all, have a good day."

WELCOME!

The Hoosier Users Group would like to welcome David Loertscher, Bob Mikels, Gary McQuade, and Galen Minton, who joined in the past month.

WELCOME BACK!

We'd also like to welcome back these renewing HUGgers: Bill Lucid, Gordon Edwards, George T. Forest, Jr., Pat McGraw, George Kruggel, Dan Eicher, Bill Cagle and Steve and Pam Sims!

HAPPY BIRTHDAY!

A Happy HUGger Birthday to these members who joined or renewed in January, 1985! Carl Walters, Jim Felton, Bill Jones of C'ville, John Cornwell, Brian Beasley, Robert Sobek, Michael Lyons, Sam Hatcher, Lew Bartley, Michael Brown and Gary Rhodes. Its time to renew!

.....DBUG.....DBBUG.....DEBUG.....

From last month's issue in the Officer's Corner, Matt McCormick should be Mack McCormick.

EDITOR'S NOTES

by Pam Sims

I don't write an editorial column on a monthly basis because I feel the space can be better utilized in printing articles, programs, tips, etc. In fact, without looking at all the back issues, I cannot remember the last time I wrote an "Editor's Column" as such. This month is an execption, however, I feel that have something important to the group, (and myself), to say.....

For the past 43 months, I have published this Newsletter for our group (sometimes at the sacrifice of my family and friends). I have not complained too much except when I'm left with no articles or workshops scheduled. Nobody knows the hours I spend each month updating our mailing lists of members and Users Groups; updating HUGbbs files and reading every Newsletter we receive in the mailbox from other Users Groups all across the country. I am indebted to Dennis Sherfy, Vic Kelson, Andy Armstrong, Bill Cagle and others who have churned out articles within an hour after my cry for help. And yes, I find all the typo's I missed after the Newsletter is printed, and last month certainly was no exception. (Murphy strikes again!)

I would like to get someone to take over the responsibility of publishing the April, 1986 Newsletter because I will not have time to do it myself. Two days after the September meeting my grandfather died. He had been ill for more than six years, his last five years were miserably spent as an invalid in a nursing home. In the first week of April, 1986, his estate auction will be held. Due to him dying without a will, the auction will be monstrous task. I would like to be available when I'm called upon to help. This is what prompted my decision to ask for a month off.

We all know, more or less, what is needed to get the Newsletter to print and to the Post Office so I won't go into it here. For those who would like to help please see me at the January and/or February Meetings.

MENGEN, A REVIEW

by Andy Armstrong

MENGEN is a utility program that will generate a machine language object file that will display all characters that you put on the screen, almost instantaneously. Its like having a compiler for menu screens!

To use the utility load the program that you wish to convert the screens of. Then find the last statement associated with the screen that you want to convert. After this line insert the statement "RUN DSK1.MENGEN". Now run the program to be converted. When the program reaches the "RUN DSK1.MENGEN" statement, the MENGEN program should load and run. While the program is running, solid blocks should appear on the screen. These solid blocks indicate where individual characters were found. After the program has "read" all the characters from the screen it will ask you for a filename (the disk to be written to must be in drive 1, just enter the filename, not DSK1.) After the you specify a filename the program will ask you for an entry point such as "MENU-1". You then have to insert in your program "CALL LOADS" and "CALL LINKS" for each screen that you converted. Once you load your converted program, access to the converted screens will be almost immediate.

THE PROGRAMMERS' CORNER

by Greg Larson

I'm announcing this month an idea for a new interest group in HUG. I'm calling it the Programmers' Corner!

This will be a group for all members who program or want to program - whatever their level. At each meeting there would be a presentation on some aspect of programming. The topic for the month will be anounced in the newsletter. A key feature will be a period for questions and answers using the resources of the group to help answer members' questions. All questions will be written down so that we can double check the answers or find answers to those we don't know. The results will be reported at the next meeting.

I've felt for some time that what we needed is a group covering programming in general, because I believe that programming problems are general in nature and not unique to any language. I was thinking that if we developed methods to support an engineering, or applied technology approach to programming, it would help generate the confidence needed for some to try programming and for others to do something ambitious.

For those just getting started in programming or learning a different language, I would like to try a "big brother" approach where more knowledgeable members would coach less experienced ones.

Possibly the most controversial idea I have is to develop programming standards for projects done in the group. People can have strong opinions about these, but whatever methods are used, they are manditory, both for others to understand your code, and for you to understand your own code next month.

The first item on the agenda is to find out the resources of the group (that's us). January is Gram-Kracker month and everyone will want to see it (me too), so I'll arrange things so that we can get in on both. I hope to make this a feature of the Programmers' Corner.

LIBRARY BITS

by Dennis Sherfy

Last month I wrote about PROW-CHART on BASIC-11. This month, I will show you how you can use the table printed by this program with the TRANSLITERATION feature of TI-WRITER.

The first thing you need to know is, why do You need TRANSLITERATION?. The Quick Reference Card which came with your TI-WRITER package list ASCII codes 32 to 127. But what about 1 to 31? And how do you get to these special characters above 127? This is where TRANSLITERATION comes in.

TRANSLITERATION is merely a long word meaning to change one key to another. If, while using TI-WRITIER, I type, .TL 65:66, this will cause my printer to print a capital B (ASCII 66) each time capital A (ASCII 65) is pressed. I have simply changed A into B. Obviously, no one would want to do this. But what if I want to enter the ELONGATED character mode on the PROWRITER? Transliteration allows me to do this. To select Elongated type, my printer requires CHR$(14), or ASCII 14. Lets go back to my original example. If I enter .TL 65:14, I will be telling my printer to convert to Elongated type each time capital A is typed. In order to get out of elongate type, my printer requires CHR$(15), or ASCII 15. I could enter .TL 66:15. This would tell my printer to convert to Elongated type when it sees an "A", then revert back to the normal type style when it encounters a "B". Now, let's get realistic. I need to use "A" and "B" frequently, so I should select two other keys that are not used so often. In my case, I do not use {-(123), }-(125), |-(124), \-(92), or `-(96) often. I could Transliterate (change) them to another character. If I enter .TL 123:14 and .TL 125:15, I will designate { and } to start and stop Elongated type. By typing Hoosier (User's) Group, the word "User's" would be in Elongated type. Transliteration is more powerful than this example. You can change a single key to represent several keys. .TL 124:27,84,51,50 will cause your Prowriter to shift to 1-1/2 line spacing when "|" is pressed. In this case "|" equals ASCII 27 (Escape), ASCII 84 (select custom line spacing), ASCII 51 (3), and ASCII 50 (2). This tells your Prowriter to change to line spacing of 32/144 inch.

To use the Greek characters or the graphic characters you must use transliteration to enter the special character mode, such as .TL 124:27,35. Then use transliteration to print each character you want.

There are two additional things you need to know to use transliteration. First, it only works with the FORMATTER. Second, you must put only one TL command on a line, followed by a carriage return.

The PRO-CHART will show you each of the available characters, and tell you which ASCII number to use with your transliteration commands.

Experiment.....and enjoy your printer and computer.

MICRO'S IN ACTION

"Getting Spritely With The 99/4A"

by Bill Cagle

I was thinking about an application for a software driven process controller. The current state of the art, has six mechanical timers, in a box that pass the timing control of the processes from one timer to the next and, at best, the timers are easy to damage. I believe this is a crude way to do business. So armed with an idea and the Extended BASIC manual I sat down and commenced to write some code. The first thing I did was to code some "bit" strings to make box es. I needed four corners, two sides, a top and a bottom line. This gave me th e characters to make a series of boxes to put the "big" sequence numbers in. The boxes were created with a loop that looks like this:

```
100 FOR I=8 TO 24 STEP 6
110 R=1 :: H=3 :: C=I+H :: W=H :: GOSUB 600
120 R=6 :: H=3 :: C=I+H :: W=H :: GOSUB 600
130 NEXT I
```

These variables are for (R)ow, (C)olumn, (W)ide, (H)igh. The sub at 600 looks like this:

```
600 CALL HCHAR(R,C,120,1) (This code prints an upper left hand corner.
610 CALL HCHAR(R,C+1,121,W) (This makes the top line)
```

The code continues till the box is printed on the screen and then it "RETURN"s to the loop to get the next set of R, C, W, and H to make the next box till all six are on the screen.

To create the "big" numbers I used code like this:

```
FOR I=1 TO 8 :: CALL CHARPAT(ASC(STR$(I)),N$(I)) :: NEXT I
```

This gets the bit pattern stored in the basic ROM and puts that pattern into th e string array called "N". Next you have to assign character numbers to the sprites with this:

```
FOR I=1 TO 6 :: CALL CHAR(I+125,N$(I)):: NEXT I
```

this assigns the character number of 126 to 132 to the "big" number sprites.

To get the big numbers on the screen, I used:

```
CALL SPRITE(#N,CHAR#,COLOR,DOT ROW, COLUMN)
```

and to get each sprite in the proper box, I used the "ON Y GOTO" statement. This was incorporated in a sub-routine like this:

```
420 GOSUB 8000

8000 ON Y GOTO 8010,8020,8030, etc.
8010 CALL SPRITE(#1,126,2,12,92) :: RETURN
8020 CALL SPRITE(#2,127,2,12,141) :: RETURN
```

this was continued until all six sprites (big numbers were defined and Y is the sequence currently being timed.

Next I had to produce a time keeping function. I used the ability of the VDP to keep a sprite in motion, once it is set in motion. This was accomplished by defining still another sprite and setting it in motion down the left side of the screen. The sprite was chosen to look like a sideways U and it slid down row of numbers. The speed was chosen by trial and error to give a speed of two lines per second. this would make the sprite pass a number every second and using the CALL POSITION(X,Y), X will have a down row value that will correspond to the number. This will let you compute the time from the value of X. I then wrote a routine that would continually check the value of X and when it was greater than the preset time for that segment, it would increment the sequence and start the next time interval. This would continue until all of the sequences are set up.

TI had made it very easy to put sprites and text on the screen at the same time. Maybe this will help some of you to start to play around with the extensive graphic command set which is built in Extended BASIC and 99/4A. If you try, you can do some tricky things with the screen and the sprites. I hope this will make you feel spritely!

This is the last of the series of FORTH corrections and updates as appeared in the August, 1985 issue of MICROpendium. Next month, Editor/Assembler changes.

BINARY SAVE OF YOUR SYSTEM DISK

Various people have given suggestions for this, starting with Craig Miller. The following may make it easier.

First, of course, decide which editor you want to use, then use it to make the changes above. Then change the definition of MENU on screen 20 to read 272 256 DO etc. This will give you garbage at the top of the screen when you first load everything, but will be useful later, as all of screen 20 can be used as the menu.

Next set up a blank screen that will load all your options in the order you wish. Here is mine:

```
( ORDER OF LOADING FOR BSAVE) BASE->R DECIMAL FORGET
-SYNONYMS
: TOM1 ; 51 LOAD 6 LOAD ( -SYNONYMS -TEXT -NEWORDS)
: TOM2 ; 57 LOAD 52 LOAD ( -GRAPH -GRAPH1)
: TOM3 ; 54 LOAD ( -GRAPH2)
: TOM4 ; 55 LOAD ( -SPLIT)
: TOM5 ; 45 LOAD ( -FLOAT)
: TOM6 ; 53 LOAD ( -MULTI)
: TOM7 ; 42 LOAD 39 LOAD 72 LOAD 89 LOAD
      ( -DUMP -COPY -PRINT -PWORDS)
: TOM8 ; 22 LOAD ( -64SUPPORT EDITOR)
: TOM9 ; ( BORDER FOR PROGRAMS WITHOUT OPTIONS)
: TOM10 ; 83 LOAD ( -BSAVE)
R->BASE
```

Notice how often my name appears (I like it). These are dummy defining words that provide borders so that I can FORGET just as much as I want. Notice that I also FORGET -SYNONYMS at the top so that words won't be duplicated. This also necessitated using xx LOAD directly, rather than the defined words. Now, once you have FLUSH'd everything, type COLD and when it is done, x LOAD, where x is the screen you used for the above, then insert a *copy* of the disk and type, in DECIMAL, ' TASK 21 BSAVE . (Include the last period.) This will print out the next available screen, and leave much of the disk for other things. Note that your original disk retains the order of loading, so that if anything happens to the BSAVE'd disk, it is easy to reconstruct. You can now edit screen 20 to take full advantage of your new MENU. My version is below, and includes other options that I placed after the BSAVE portion.

```
ALREADY LOADED: [FORGET BACK TO (XXX)]
(TOM1) -SYNONYMS -TEXT -NEWORDS
(TOM2) -GRAPH -GRAPH1   (TOM3) -GRAPH2
(TOM4) -SPLIT           (TOM5) -FLOAT
(TOM6) -MULTI (TOM7)-DUMP -COPY -PRINT
(TOM8) -64SUPPORT    - (TOM9) options
(TOM10) free, re-enter if forgotten
New words: forget with TOM1
BASEINDEC EB PAGE SIZE NEW BYE-flushes  2FORMAT-DISK
Printer words: loaded with -PRINT
P" PCR PLIST PVLIST PDUMP PINDEX
Available options:
-ASSEMBLER -CRU -BSAVE -TRACE -TESTHEAD -FORTHTRAN
-DOUBLECOPY (clears all memory first)  -DECOMPILE
-SEARCH -2FORTHCOPY
```

Note that I indicate how to FORGET each section. Also, anything beyond the 40th character on a line will be printed on a new line on your screen, so arrange this screen carefully.

Now all you need to do is edit screen 3 to BLOAD what you have. Here is mine:

```
( WELCOME SCREEN ) BASE->R HEX F0 7 8 SYSTEM 10 SYSTEM
0 0 GOTOXY ." Loading...TI FORTH" 10 83C2 C! DECIMAL 21 BLOAD
FORGET TOM10 ( eliminate BSAVE)
: -ASSEMBLER 34 LOAD ; : -TRACE   50 LOAD ; : -CRU   42 LOAD ;
: -FORTHTRAN 46 LOAD ; : -TESTHEAD 44 LOAD ; : -BSAVE 43 LOAD ;
: -DOUBLECOPY 48 LOAD ; : -DECOMPILE 51 LOAD ;
: -SEARCH 55 LOAD ; : -2FORTHCOPY 33 LOAD ;
1 VDPMDE ! 0 DISK_LO ! 180 DISK_SIZE ! 540 DISK_HI !
PAGE MENU  : TOM10 ; ( BORDER PRESERVING OPTIONS)
R->BASE
```

Now FLUSH again and you are ready. If you type COLD, you should get the whole thing back, with a nice neat menu on the whole screen.

SHORT FORTHTRAN

Here is the shortest FORTHTRAN I know (mainly because all comments are eliminated). Apologies to Mike Amundsen for this one, but it is modified to allow multiple screen transfers at once. When it is loaded the first time brief instructions are written on the screen. Note that in order to use quotes on screen I had to make a new word, appropriately called ", since this can't be put inside of ."

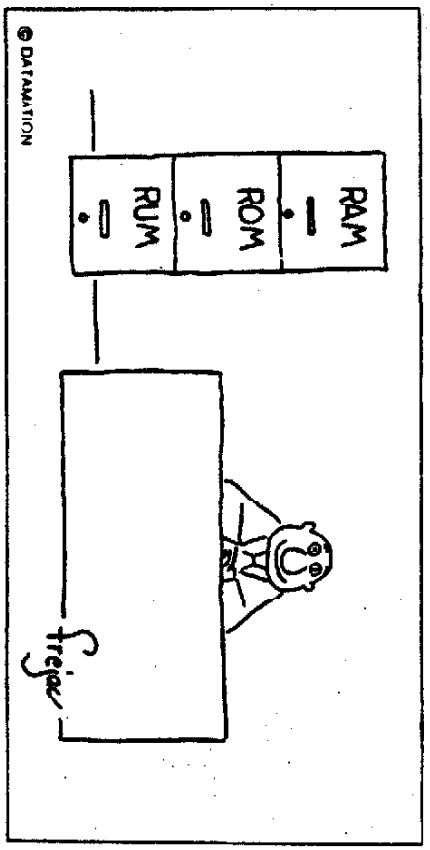
```
( SHORT FORTHTRAN (DEC84 TSF) BASE->R DECIMAL
                : " 34 EMIT ; : INS1 CLS 0 9 GOTOXY ." ENTER
 FIRST SCREEN #, NUMBER OF SCREENS  TO MOVE, THEN:" CR CR ." DSK
-SCR OR SCR-DSK" ; HEX : INS2 ." TYPE INS1 TO REPEAT OR FORGET "
 " ; 0 VARIABLE FILBUF 50 ALLOT PABS 2 A + FILBUF 1900 FILE FILT
RAN  : SETFILE FILTRAN SET-PAB SWNTL DSPLY VRBL 50 REC-LEN ;
 DECIMAL     : PUTFILE   OUTPT FILTRAN OPN OVER + SWAP DO I
     BLOCK                 CR CR ." FILE TRANSFER IN PROGRESS..
" CR CR 16 0 DO DUP FILBUF 64 CMOVE I REC-NO 64 WRT 64 + LOOP
 DROP LOOP CLSE BEEP ." DISK FILE COMPLETED. " CR CR INS2 QUIT ;
 : GETFILE INPT FILTRAN OPN OVER + SWAP DO I DUP BLOCK
  CR CR ." FILE TRANSFER IN PROGRESS..."
  CR CR 16 0 DO FILBUF 64 BLANKS
I REC-NO RD DROP DUP FILBUF SWAP 64 CMOVE 64 +
 LOOP DROP BLOCK DROP UPDATE LOOP CLSE FLUSH BEEP
 ." SCREEN COMPLETED"  CR CR INS2 QUIT ;      R->BASE      -->
( SHORT FORTHTRAN, P.2)  BASE->R
: INS ." ENTER FILE DESCRIPTOR WORD-USE THE FORM" CR
." F-D" " ."  DSKx.XXX" " CR CR ." THEN TYPE " ;
: SCR-DSK   CLS 9 3 GOTOXY ." SCREEN TO DISK TRANSFER"  9 4
   GOTOXY ." ==========="
  CR CR CR CR CR SETFILE BEEP INS " ." PUTFILE" '  CR QUIT ;
: DSK-SCR   CLS 9 3 GOTOXY ." DISK TO SCREEN TRANSFER"  9 4
   GOTOXY ." ==========="
  CR CR CR CR CR SETFILE BEEP INS " ." GETFILE" '  CR QUIT ;
INS1     R->BASE
```

Download characters to your Gemini

The following program was written by Jim Peterson of Columbus, Ohio. Many TI users know Peterson as the author of the Tips from Tigercub column that appears regularly in many user group newsletters.

The program, called DOWNCHAR, permits on-screen design of downloadable characters for Gemini printers. It is also compatible with Epson printers. The program features a direct dump to the printer for viewing the newly designed character and optional saving to disk. Peterson released the program to the public domain.

```
100 CALL CLEAR :: CALL SCREE
N(4):: CALL CHAR(128,"FF8181
81818181FF",129,RPT$("F",16)
):: CALL COLOR(13,2,16)
110 FOR R=9 TO 15 :: CALL HC
HAR(R,11,128,9):: NEXT R
120 X=1 :: FOR R=9 TO 15 ::
DISPLAY AT(R,7)SIZE(2):STR$(
X):: X=X*2 :: NEXT R :: FOR
C=9 TO 17 :: DISPLAY AT(8,C)
SIZE(1):STR$(C-8):: NEXT C
130 DISPLAY AT(2,9):"TIGERCU
B'S" :: DISPLAY AT(4,1):"GEM
INI CHARACTER DOWNLOADER" !p
rogrammed by Jim Peterson fo
r the Public Domain
140 DISPLAY AT(17,1):" Move
cursor with W,E,R,S,D,":"Z,X
 and C keys. Toggle on":"and
 off with Q key. Press":"Ent
er when finished.": : :"Pres
s any key"
150 CALL KEY(0,K,ST):: IF ST
=0 THEN 150 :: CALL HCHAR(17
,1,32,224)
160 R=9 :: C=11 :: CH=128
170 CALL HCHAR(R,C,32):: CAL
L HCHAR(R,C,CH):: FOR D=1 TO
 10 :: NEXT D :: CALL KEY(3,
K,ST):: IF ST=0 THEN 170
180 ON POS("QWERDCXZS"&CHR$(
13),CHR$(K),1)+1 GOTO 170,31
0,230,220,210,200,190,260,25
0,240,330
190 R=R+1
200 C=C+1 :: GOTO 270
210 C=C+1
220 R=R-1 :: GOTO 270
230 R=R-1
240 C=C-1 :: GOTO 270
250 C=C-1
260 R=R+1
270 R=R-(R<9)+(R>15):: C=C-(
C<11)+(C>19):: IF CH=128 THE
N 300 :: CALL GCHAR(R,C-1,GX
):: CALL GCHAR(R,C+1,GZ):: I
F (GX<>129)*(GZ<>129)THEN 30
0
280 DISPLAY AT(22,1):"You ca
n't have two in a row":"hori
zontally!" :: FOR D=1 TO 50
:: NEXT D :: DISPLAY AT(22,1
):" ":" "
290 CH=CH-1
300 CALL HCHAR(R,C,CH):: GOT
O 170
310 CH=CH+1+(CH=129)*2 :: IF
 CH=128 THEN 320 :: CALL GCH
AR(R,C-1,GX):: CALL GCHAR(R,
C+1,GZ):: IF (GX<>129)*(GZ<>
129)THEN 320 ELSE 280
320 CALL HCHAR(R,C,CH):: GOT
O 170
330 FOR C=11 TO 19 :: X=1 ::
 FOR R=9 TO 15 :: CALL GCHAR
(R,C,G)
340 IF G=129 THEN A=A+X
350 X=X*2 :: NEXT R
360 FOR J=1 TO LEN(STR$(A)):
: CALL VCHAR(15+J,C,ASC(SEG$
(STR$(A),J,1))):: NEXT J ::
M$=M$&CHR$(A):: A=0 :: NEXT
C :: A=0
370 DISPLAY AT(20,1):"Print?
 Y/N Y" :: ACCEPT AT(20,12)V
ALIDATE("YN")SIZE(-1):Q$ ::
IF Q$="N" THEN 470
380 IF F=1 THEN 390 :: F=1 :
: DISPLAY AT(20,1):"Printer
name?" :: ACCEPT AT(20,15):P
$ :: OPEN #1:P$
390 DISPLAY AT(20,1):"ASCII
to redefine?" :: ACCEPT AT(2
0,20)VALIDATE(DIGIT)SIZE(3):
CH
400 DISPLAY AT(20,1):"Descen
der (0 or 1)? 0" :: ACCEPT A
T(20,21)VALIDATE("01")SIZE(-
1):D$ :: D=VAL(D$)
410 M$=CHR$(27)&CHR$(42)&CHR
$(1)&CHR$(CH)&CHR$(D)&M$
420 PRINT #1:M$ :: PRINT #1:
CHR$(27);CHR$(36);CHR$(1);
430 PRINT #1:RPT$(CHR$(CH),7
2):: PRINT #1:CHR$(14);RPT$(
CHR$(CH),36)
440 DISPLAY AT(20,1):"Save (
Y/N)? Y" :: ACCEPT AT(20,13)
VALIDATE("YN")SIZE(-1):Q$ ::
 IF Q$="N" THEN 470
450 IF F3=1 THEN 460 :: F3=1
 :: DISPLAY AT(20,1):"Filena
me? DSK" :: ACCEPT AT(20,14)
:F$ :: OPEN #2:"DSK"&F$
460 PRINT #2:M$
470 M$="" :: DISPLAY AT(20,1
):"Another (Y/N)? Y" :: ACCE
PT AT(20,16)VALIDATE("YN")SI
ZE(-1):Q$ :: IF Q$="Y" THEN
100
480 CLOSE #1 :: CLOSE #2 ::
END
```



TIPS FROM THE TIGERCUB

827

TIGERCUB SOFTWARE
156 Collingwood Ave.
Columbus, OH 43213

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

The entire contents of Tips from the Tigercub Nos. 1 through 14, with more added, are available as a full disk of 50 programs, routines and files for just $15.00 postpaid.

>>>>>>NOW AVAILABLE<<<<<<<<<

Tips from the Tigercub VOLUME 2 The entire contents of Tips Nos. 15 through 24, with 64 routines and files, also $15.00 postpaid.

Or both for $27 postpaid.

>>>>ALSO NOW AVAILABLE<<<<<<

New Catalog #6, for $1 which is deductable from your first order. Describes 140 original programs for only $3 each (plus $1.50 per order for casette or disk, package and postage).

If you have my previous catalog, the following are now available in Extended Basic versions - Fast Addition Practice, Submarine Hunt, Ritmatik, Wawaland (also now available in Basic with Speech), Long Division Cryptograms, Miss Spell, Scramabulation, Bargraffer, Squinch, Dry Gulch, Name That Tune, Scrum, Midnight Trail, Nimbo, Kindertimes, Optical Illusion, Bazoo, Synonyms, Speeder Reader, Changeroo, Glunk, Fraction Math, Three Buckets Puzzle, Roman Numbers, Match A Patch, Kinderminus, I & E Spelling, Casting Out Nines, Haunted Graveyard, Spelling Teacher, Homonyms, Antonyms, Old -Timer Puzzle, Ten Thousand Sights, Mechanical Aptitude Test, Junior Speeder Reader, and Bars and Balls.

Due to reduced prices for disks and mailers, the PPH charge is now $1.50 for either disk or casette - BUT PLEASE BE SURE TO SPECIFY WHICH!

And my best seller - NUTS & BOLTS, a full disk of 100 (yes, I said 100) utility subprograms in MERGE format, ready for you to merge into your own programs. 13 type fonts, 14 text display routines, 9 wipes, 8 pauses, 3 programming aids, 9 data saving and reading routines, 5 graphics routines, 4 time and date, 6 music, 12 sorts and shuffles, 2 printer aids, 4 key and joystick, 4 math, 2 protection and 7 miscellaneous, plus a tutorial on subprograms. With documentation, example of using each subprogram. All for only $19.95 postpaid.

I have been receiving several requests to publicize freeware which is sometimes good but sometimes of doubtful usefulness, quality, originality or even legality! And "Freeware! Send $10 and initialized disk" is not freeware, it's somebody trying to get a free ad! So - no more freeware mentions! I am also not going to mention commercial products - after all, I'm publishing this at my own expense to promote my own software! However, I do owe a mention to Larry Hughes of Quality 99 Software, because in Tips #22 I recommended that disks with fractured files should not be copied with a quick copier. Larry informed that his trademarked QUICK-COPYer is the only program of its kind on the market that does un-fracture files. He sent along a copy to prove it, and it does just that. A very useful feature!

Now, here is the new, and final, version of the Tigercub Menu Loader.

```
100 !by A. Kludge/M. Gordon/
T. Boisseau/J. Peterson/etc.
Version #5, 9/85
110 CALL PEEK(8198,A):: IF A
<>170 THEN CALL INIT
120 OPTION BASE 1 :: DIM PG$
(127),V(127,3):: CALL LOAD(-
31806,16):: ON ERROR 130 ::
GOTO 160
130 DISPLAY AT(12,9)ERASE AL
L:"I/O ERROR" :: RUN 100
140 @,@@,A,A$,B,C,D$,FLAG,I,
J,K,KD,KK,M,M$,N$,NN,P,P$,PG
$(),PP,PP$,Q$,S,ST,T$(),TT,V
T,V(,),W$,X,X$,Y,K2,S2
150 CALL LINK :: CALL PEEK :
: CALL KEY :: CALL SCREEN ::
 CALL COLOR :: CALL CLEAR ::
 CALL VCHAR :: CALL SOUND ::
 !@P-
160 CALL CLEAR :: CALL LOAD(
8196,63,248):: CALL LOAD(163
76,67,85,82,83,79,82,48,8)
170 CALL LOAD(12288,129,195,
126,165,129,153,102,60)
180 CALL LOAD(12296,2,0,3,24
0,2,1,48,0,2,2,0,8,4,32,32,3
6,4,91):: CALL LINK("CURSOR"
)
190 CALL CLEAR :: FOR S=1 TO
 14 :: CALL COLOR(S,7,16)::
NEXT S :: CALL COLOR(0,2,16)
200 T$(1)="d/f" :: T$(2)="d/
v" :: T$(3)="i/f" :: T$(4)="
i/v" :: T$(5)="pro" :: ON WA
RNING NEXT
210 IMAGE ###
220 IMAGE ### Quit
230 IMAGE ### Delete
240 IMAGE ### Print
250 IMAGE ### Rescan
260 CALL SCREEN(5):: CALL VC
HAR(1,31,1,96):: DISPLAY AT(
1,4):"TIGERCUB MENU LOADER"
270 ! IF YOU HAVE MORE THAN
ONE DISK DRIVE, DELETE THE !
 IN LINE 280 AND THE FIRST S
TATEMENT IN 210
280 ! DISPLAY AT(12,6):"DISK
? (1-3):" :: ACCEPT AT(12,19
)SIZE(-1)VALIDATE("123"):D$
:: D$="DSK"&D$&"."
290 D@="DSK1." :: OPEN #1:D@
,INPUT ,RELATIVE,INTERNAL ::
 INPUT #1:N$,A,J,K :: DISPLA
Y AT(1,2)SIZE(27):SEG$(D@,1,
4)&" - Diskname= "&N$;
300 DISPLAY AT(2,2):"Availab
le=";K;"Used=";J-K;" Prog Fi
lename Size  Type":"---- ---
------- ---- -------" :: I,V
T=0 :: TT=J-K
310 FOR X=1 TO 127 :: IF X/2
0<>INT(X/20)THEN 340
320 DISPLAY AT(24,1):"Choice
? Enter for more 0" :: ACCEP
T AT(24,24)VALIDATE(DIGIT)SI
ZE(-3):K :: IF K=0 THEN 330
:: IF K>0 AND K<NN+1 THEN 60
0 ELSE 320
330 X=1
340 I=I+1 :: IF I>127 THEN K
=X :: GOTO 510
350 INPUT #1:P$,A,J,B :: NN=
NN+1
360 IF LEN(P$)=0 THEN 430
370 DISPLAY AT(X+4,1):USING
210:NN :: DISPLAY AT(X+4,5):
P$ :: PG$(NN)=P$ :: DISPLAY
AT(X+4,16):USING 210:J :: DI
SPLAY AT(X+4,20):T$(ABS(A))
380 V(NN,1)=A :: V(NN,2)=ABS
(B):: V(NN,3)=J
390 X$="  "&STR$(B):: DISPLA
Y AT(X+4,24):SEG$(X$,LEN(X$)
-2,3):: VT=VT+J
400 IF A>0 THEN 410 :: DISPL
AY AT(X+4,28):"Y"
410 CALL KEY(0,KK,ST):: IF S
T=0 THEN 420 :: FLAG=1 :: GO
TO 430
420 NEXT X
430 DISPLAY AT(X+4,1):USING
220:NN :: DISPLAY AT(X+5,1):
USING 230:NN+1
440 IF VT=TT OR FLAG=1 THEN
460 :: DISPLAY AT(2,25)SIZE(
4):VT
450 FOR @=1 TO 10 :: DISPLAY
 AT(2,25)SIZE(1):CHR$(30)::
DISPLAY AT(2,25)SIZE(1):" "
:: CALL SOUND(-99,110,0,-4,0
):: NEXT @
460 IF FLAG=1 THEN 470 :: DI
SPLAY AT(X+4,13):USING 240:N
N+2 :: DISPLAY AT(X+5,13):US
```

```
ING 250:NN+3
470 DISPLAY AT(X+6,1):"    C
hoice?" :: ACCEPT AT(X+6,16)
SIZE(-3)VALIDATE(DIGIT):K
480 IF FLAG=1 THEN 500
490 IF K=NN+2 THEN 840 ELSE
IF K=NN+3 THEN CLOSE #1 :: N
N=0 :: GOTO 190
500 IF K<>NN AND K<>NN+1 THE
N 590
510 IF K=NN THEN CALL CLEAR
:: CLOSE #1 :: END
520 DISPLAY AT(X+5,12)SIZE(1
2):" #?" :: ACCEPT AT(X+5,15
)SIZE(2)VALIDATE(DIGIT):KD :
: IF KD<1 OR KD>NN THEN 520
530 IF V(KD,1)>0 THEN 550
540 FOR J=1 TO 10 :: DISPLAY
 AT(11,1):" ":" PROTECTED -
 CANNOT DELETE":" " :: DISPL
AY AT(12,1):" " :: NEXT J ::
 GOTO 370
550 DISPLAY AT(X+6,1)SIZE(27
)BEEP:" Verify - Delete ";P6
$(KD);"?" :: DISPLAY AT(X+6,
28)SIZE(1):"Y" :: ACCEPT AT(
X+6,28)SIZE(-1)VALIDATE("YN"
):Q$ :: IF Q$<>"Y" THEN 570
560 DELETE D$&PG$(KD)
570 CLOSE #1
580 CALL VCHAR(1,3,32,672)::
 NN=0 :: X=0 :: FLAG=0 :: GO
TO 260
590 IF K<1 OR K>127 OR LEN(P
G$(K))=0 THEN 430
600 IF ABS(V(K,1))=5 OR ABS(
V(K,1))=4 AND V(K,2)=254 THE
N 640
610 DISPLAY AT(12,1)ERASE AL
L:"Print to ? S": :"(P)printe
r?":"(S)creen?" :: ACCEPT AT
(12,12)SIZE(-1)VALIDATE("PS"
):Q$ :: IF Q$="S" THEN PP=0
:: GOTO 630
620 DISPLAY AT(12,1)ERASE AL
L:"PRINTER? PIO" :: ACCEPT A
T(12,10)SIZE(-18):P$ :: OPEN
 #3:P$ :: PP=3
630 CALL CLEAR :: CALL SCREE
N(16):: ON ABS(V(K,1))GOTO 6
80,690,750,760
640 CLOSE #1 :: IF SEG$(PG$(
K),LEN(PG$(K)),1)="*" THEN D
ISPLAY AT(12,1)ERASE ALL:"RE
TURN TO BASIC AND LOAD BY":"
TYPING OLD ";D$&PG$(K):: STO
P
650 CALL PEEK(-31952,A,B)::
CALL PEEK(A*256+B-65534,A,B)
:: C=A*256+B-65534 :: A$=D$&
PG$(K):: CALL LOAD(C,LEN(A$)
)
660 FOR I=1 TO LEN(A$):: CAL
L LOAD(C+I,ASC(SEG$(A$,I,1))
):: NEXT I :: CALL LOAD(C+I,
0)
670 CALL VCHAR(1,3,32,672)::
 CALL SCREEN(8):: FOR S=8 TO
 14 :: CALL COLOR(S,2,1):: N
EXT S :: DISPLAY AT(12,2):"L
OADING ";A$ :: GOTO 900
680 OPEN #2:D$&PG$(K),INPUT
,FIXED :: GOTO 700
690 OPEN #2:D$&PG$(K),INPUT
700 LINPUT #2:M$ :: PRINT #P
P:M$ :: IF EOF(2)THEN 730
710 CALL KEY(8,K,S):: IF S=0
 THEN 700
720 CALL KEY(8,K2,S2):: IF S
2<1 THEN 720 ELSE 700
730 CLOSE #1 :: CLOSE #2 ::
PRINT "   >>>press any key<<
<" :: IF Q$="P" THEN CLOSE #
3
740 CALL KEY(8,K,ST):: IF ST
<1 THEN 740 ELSE 580
750 OPEN #2:D$&PG$(K),INPUT
,INTERNAL,FIXED :: J=8 :: GO
TO 770
760 OPEN #2:D$&PG$(K),INPUT
,INTERNAL :: J=8
770 IF EOF(2)=1 THEN 730 ::
J=J+1 :: INPUT #2:M$ :: IF L
EN(M$)=8 THEN 790
780 PRINT #PP:M$ :: GOTO 820
790 FOR Y=1 TO 8 :: @@=ASC(S
EG$(M$,Y,1)):: IF @@<32 OR @
@>127 THEN 810
800 NEXT Y :: GOTO 780
810 RESTORE #2 :: FOR X=1 TO
 J-1 :: INPUT #2:M$ :: NEXT
X :: INPUT #2:M :: PRINT #PP
:M
820 CALL KEY(8,K,S):: IF S=0
 THEN 770
830 CALL KEY(8,K2,S2):: IF S
2<1 THEN 830 ELSE 770
840 DISPLAY AT(24,1):"PRINTE
R NAME? PIO" :: ACCEPT AT(24
,15)SIZE(-14):PP$ :: OPEN #2
:PP$ :: PRINT #2:SEG$(D$,1,4
)&" - Diskname= "&N$
850 PRINT #2:RPT$("=",28):"A
vailable=";358-VT;"Used=";VT
:RPT$("=",28)
860 PRINT #2:"FILENAME  SIZE
   TYPE":RPT$("-",28)
870 FOR P=1 TO NN-1 :: PRINT
 #2:P6$(P);TAB(15);V(P,3);TA
B(20);T$(ABS(V(P,1)));TAB(25
);V(P,2):: NEXT P :: CLOSE #
2
880 DISPLAY AT(12,3)ERASE AL
L:"(P) to print again":" (R
) to rescan":" (Q) to quit"
890 ACCEPT AT(15,4)VALIDATE(
"PQR")SIZE(-1)BEEP:Q$ :: IF
Q$="P" THEN 840 :: CLOSE #1
:: NN=0 :: IF Q$="R" THEN 19
0 ELSE END
900 RUN "DSKX.1234567890"
```

This version turns off the Quit key, restarts itself rather than crashing on an I/O error, and has pre-scan for faster start-up. It displays disk name, sectors available and sectors presumably used - it also totals up actual sectors used and sounds a warning if any sectors are not accounted for.

It lists up to 127 programs and files by number, filename, number of sectors, program or file type, file record length, and write-protection. It will stop for menu selection on any keypress or at the end of each screen, continuing on Enter. It will load and run any program that can run from Extended Basic, displaying its filename while loading. If the filename ends in an asterisk, it will warn you to return to Basic. It will delete any unprotected program or file, after first requiring verification by filename, or will inform you if the file is protected. It will read any readable file, including internal numeric, and list it to screen or printer. It will dump a catalog of the disk to your printer, and it will offer the option of quitting or rescanning the disk or another disk. And it's free, I don't even want a freeware donation - but I would appreciate if you would take a look at my catalog and see if somewhere among those 140 programs, there might be something you would be willing to pay $3 for? The Menu Loader is included as a bonus on every disk I sell!

```
100 CALL CLEAR :: RANDOMIZE
:: DISPLAY AT(3,4):"TIGERCUB
 MATH PUZZLE"
110 DISPLAY AT(6,1):"Insert
+, -, * (multiply) OR / (div
ide) between the  digits
to equal the total": :"Type
0 to give up"
120 DISPLAY AT(12,1):"Level
1 or 2?" :: ACCEPT AT(12,15)
VALIDATE("12"):L$
130 T,X=INT(9*RND+1):: M$=ST
R$(X):: Z$=M$&" "
140 FOR J=1 TO 4 :: Y(J)=INT
(9*RND+1):: Z=INT(4*RND+1)::
 ON Z GOSUB 240,250,260,270
:: Z$=Z$&STR$(Y(J))&" " :: N
EXT J
150 IF L$="1" AND T<>INT(T)T
HEN 130 :: Z$=Z$&"="&STR$(T)
160 DISPLAY AT(12,1):Z$ :: D
ISPLAY AT(18,1):" " :: DISPL
AY AT(20,1):" " :: DISPLAY A
T(22,1):" "
170 P=2 :: FOR J=1 TO 4 :: A
CCEPT AT(12,P)VALIDATE("Q+-*
/")SIZE(1):S$
180 IF S$="Q" THEN 200 ELSE
IF S$="+" THEN X=X+Y(J)ELSE
IF S$="-" THEN X=X-Y(J)ELSE
IF S$="*" THEN X=X*Y(J)ELSE
X=X/Y(J)
190 P=P+2 :: NEXT J :: IF X=
T THEN 230 :: DISPLAY AT(18,
1):"WRONG!"
200 DISPLAY AT(20,1):"ANSWER
 IS ";M$
210 DISPLAY AT(22,1):"PRESS
ANY KEY"
220 CALL KEY(8,K,ST):: IF ST
<1 THEN 220 :: GOTO 130
230 DISPLAY AT(18,1):"RIGHT!
" :: GOTO 210
240 M$=M$&"+"&STR$(Y(J)):: T
=T+Y(J):: RETURN
250 M$=M$&"-"&STR$(Y(J)):: T
=T-Y(J):: RETURN
260 M$=M$&"*"&STR$(Y(J)):: T
=T*Y(J):: RETURN
270 M$=M$&"/"&STR$(Y(J)):: T
=T/Y(J):: RETURN
```

Enjoy!

Jim Peterson

HOOSIER USERS GROUP DIRECTORY

HOOSIER USERS GROUP OFFICERS

President.....Steve Sims 631-7255
Vice-President.....Bill Lucid 291-3995
Secretary.....Barb Uhrig 357-8268
Treasurer.....Bill Jones

COMMITTEE CHAIRPERSONS

Regional Centers:
South.....Dennis Sherfy 881-5918
West Indiana..Vic Kelson 812-234-5533

Documents.....Don Donlan 882-4544
Membership.....Pam Sims 631-7255
Newsletter.....Pam Sims 631-7255

MONTHLY MEETING LOCATION

Creative Logic
8240 Indy Lane
Indianapolis, IN 46224

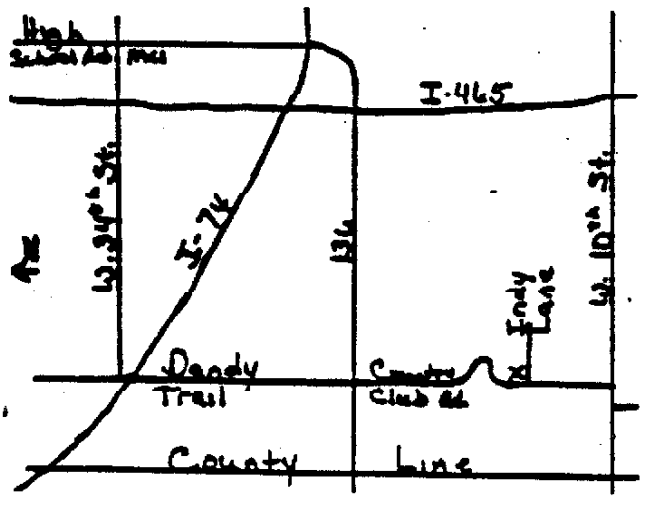
(About 1800 North Country Club Road)

NEWSLETTER EXCHANGE

The Hoosier Users is participating in a Newsletter Exchange program with other TI Users Groups. This offer is made with the understanding that, with proper credit, your Users Group can reprint articles from the Hoosier Users Group Newsletter, and with proper credit, we can reprint articles from other TI Users Groups Newsletters.

PRINTOUTS

Library listings can be ordered for $.25 & a 6x9 self addressed envelope with $.66 postage. The HUGbbs Reference Guide can be ordered for $.50 and a 4x9 self addressed envelope with $.22 postage. Please send orders to our P.O. Box. SORRY, PRINTOUTS WILL BE SENT TO ACTIVE MEMBERS ONLY!



HUGbbs INFORMATION

317-631-994A

The HUGbbs operates on a 24 hour basis.

SPONSOR THE HUGbbs: Any member or retail business can sponsor the HUGbbs. For a $5.00 donation, you get 5 (40 column) lines on the Log-On Title Screen for a week (or for a $10.00 donation, you get 10 (40 column) lines) plus a 24 line by 40 character ad in the Sales option of the File Module. To sponsor the HUGbbs, send a check or money order to our P.O. Box (or turn in at our Monthly Meeting) specifying how many weeks (and how many lines) you want to sponsor, your name (or company name), address, phone, what you want to say, and the week (and an alternate week) you want the ad to appear.*

BACK ISSUES

Back Issues purchased at the monthly meeting is $1.00 each. Mail order price is $1.50 per Newsletter (postage included). Orders will be filled within 3 weeks of receipt by the Documents Committee.

ADVERTISING POLICIES

There will be no charge for advertisements submitted to the HUGger Newsletter by members (for private sale only). Format for the advertisements is 45 characters wide by 10 lines long. The Ad should be typed or hand printed exactly how it is to appear in the Newsletter. Deadline for an ad to appear in next month's Newsletter is the 2nd Saturday of the month.*

For companies who wish to advertise in the HUGger Newsletter, our rates are as follows:
Pre-Printed Inserts (one page) $20.00
One Full Page (one sided) Ad: $25.00
One Half Page Ad: $13.00
One Quarter Page Ad: $7.00
All ads must be in a ready to print condition. Advertisements must be in our P.O. Box before the 2nd Saturday of the month to appear in the following month's Newsletter.*

***NOTE:** The Officers of the Hoosier Users Group reserve final approval on all advertisements submitted for the HUGger Newsletter and the HUGbbs. The Officers and the Newsletter committee are not responsible for typographical errors due to illegible advertisements. All proceeds are accepted as donations to the Hoosier Users Group.

JAN. 7 1986

TIME DATED
JAN. 7 1986
MATERIAL

HOOSIER USERS GROUP
P.O. Box 2222
Indianapolis, IN 46206-2222

Forwarding and Address
Correction Requested

Bulk Rate
U.S. Postage
PAID
Indianapolis, IN
Permit No. 6440

APPLICATION FOR MEMBERSHIP

Below you will find an application for membership to the Hoosier Users Group. Active membership entitles you to the Newsletter, up and download on the HUGbbs, attendance and voting rights at regular club meetings, access to the HUGger Library of Programs, special club activities and special guest speakers for one year. Subscribing members will receive the **NEWSLETTER** only.

Make check or money order payable to **Hoosier Users Group.** Send completed application to:

HOOSIER USERS GROUP
P.O. Box 2222
Indianapolis, IN 46206-2222

(Cut on dotted line)

Check One:

Active Member

New: $20 ____
Renewal: 15 ____

Subscribing Member

New: $10 ____
Renewal: 7.50 ____

Amount Enclosed: $ ____

____ D ____
S ____ O ____

Name:_____ Today's Date:_____

Address: _____ Apt. # ____

City:_____ State:____ Zip:_____

Phone: (____) ____-____

Interests/Comments: _____

_____