Portland
Nov 86

THE PUNN NEWSLETTER
Wordplay
P.O.BOX 15037
PORTLAND, OREGON 97215

WORDPLAY VOL. V, NO. 11
NOVEMBER 1986

NOMINATIONS REPORTED

At the October meeting of the PUNN board the Nominatons Committee reported their recommendations for 1987 PUNN Officers:

PRESIDENT: DUANE GOODMAN. Duane has served in various capacities including Treasurer, Chairman of BBS Committee, and his current position of Vice President

VICE PRESIDENT: KEITH FAST. Keith has also served in several positions, most recently as our current Treasurer, where he has carried on the fine level of services established by Duane

TREASURER: MIKE KING. A member over two years, Mike has contributed an enormous amount of time and effort the programming of the PUNN BBS, and is active in workshops.

SECRETARY: DON BARKER. Don is retired and has much time and energy to devote to getting things done for the Users Group, and is willing to serve.

The committee feels this is a well balanced ticket. These nominees, as well as nominations (or volunteers) from the floor will be presented at the November 4th PUNN meeting. Elections will be held at the December 2nd meeting.

Our current President, Cricket Raybern, who has served us so well over the past two years has agreed to serve in the capacity of Director-at-Large, for which we thank her.

QUICK GLIMPSES OF THE SEATTLE FAIR

BY AL KINNEY

* COMPUTE! magazine is dropping its rather meager coverage of the TI. Cheryl (REGENA) Whitlaw announced that she would continue to write and market her own programs, but on a much smaller scale than in the past. She is going to devote more time to family.

* CHRIS BOBBIT of ASGARD software is working steadily to release new software for the TI.

* CRAIG MILLER of MILLER GRAPHICS made an announcement, of sorts, about a new project he is working on. This is supposed to be something that will bring FULL IBM PC compatibility to your lowly TI-99/4A ... hardware and software. I guess I'm in the awkward stage. I know just enough about computers to know that is gonna be real difficult, given the differences in hardware and software. Otherwise, maybe I don't know enough, and it will be *THE* product everyone has been waiting for...

* LOU PHILLIPS of MYARC had a working prototype of his much discussed, little understood GENE'VE 99/4A compatible. The news <for me> is that I really was impressed, and hope that it becomes fact and doesn't fizzle! The machine will bring a whole new dimension to the term 'home computer.' It is intended to be a complete package, with nothing extra to buy. That means you will receive the board, keyboard and necessary software and documentation to be able to use it immediately upon receipt. The critical question is *STILL*, "WHEN?" It is now a month since the Seattle Fair, and the amount of traffic on COMPUSERVE in regard to it, is underwhelming! Lou really expected to be shipping by now, I believe, the only delay being a complex Gate Array, that is the heart of the MYARC operating system. Lou candidly admitted that shipping for the upcoming Christmas shopping period was an absolute must!

* JIM HORN, System Operator (SYSOP) of the TI Forum on Compuserve announced the formation of the FIRST Gene've Users Group. Talk about being optomistic!! The stated purpose of the group is to promote use and support for the new MYARC machine, but anyone will be welcome to participate.

If you didn't go, you shoulda'!!

PRBASE VERSION 2.0

BY DUANE GOODMAN

Not having used PRBASE, Version 1.0, I am unable to point out the differences between 1.0 and 2.0 other than what is included in the documentation. So this review will be mostly on what 2.0 will do for you.

PRBASE comes on two SSSD diskettes. Disk one has the programs needed for setting up (CREATE) and running (PRBASE) the program. There are 15 files in this group for a total of 353 sectors. Disk two consists of one file which contains all of the documentation for PRBASE. This file is 357 sectors long. Included on disk one is a file called PRINTFILE which will print out the data file.

The first thing you will want to do is make a backup of both disks. Then print the documentation file, as you will need to read it over before you start to set up your data base and you will refer to it many times in the process.

Equipment needed for PRBASE is as follows.

TI 99/4A with 32K
Single Disk Drive (two work better!)

and one of the following cartridges:

X-BASIC
Editor Assembler

Mini Memory
TI-Writer

Optional, but needed to make full use of the system is an RS232 interface and printer.

After loading the program you are presented with a 3 item menu.

1. Is the creation menu.
2. Is the data base management
3. Is to exit back the the TI title screen.

The first time through you will be required to select option one of this menu so that you may design your data entry screen and your report formats. The creation menu looks like this:

CREATION MENU

1. to Select Data Disk Drive
2. Format Data Diskette
3. Design Data Screen
4. Design Tabular Reports
5. Design Mailing Labels
6. Set Printer Codes
7. Set System Options
8. Exit

Option one on this menu will let you tell the program which drive your data disk is in.

Option two formats the data disk. PRBASE uses a non-standard format for the data disk which is not readable with the DOS. To back up or copy your data disk, you will need to use a "sector" copy program. MASSCOPY works fine, as does the sector copy of DM1000. Also included with PRBASE is a utility program which will do this (more on that later).

Option three is where you will do most of your work. This is where you design the format of your data input screen. You are presented with a list of special characters while designing your data screen which serve to your data screens more appealing to the eye. Fields are marked off by typing paired brackets or braces where you wish your data to appear when using the main PRBASE program. You may have up to a total of 32 fields which total no more than 255 characters. This is "one" data record. A SSSD system will hold 350 data records. Double sided diskettes hold as many as 710 records.

Option four is where you design you tabular reports. You may have up to 5 different tabular report forms. This allows you to set up your output for things such as Rolodex cards, tabular reports such as our one across mailing list for the club. You can also have this information go to a disk file which will allow you to create a mail merge file for use with the mail merge option of TI-WRITER. PRBASE opens the disk file in the UPDATE mode which allows you to make multiple searches and output them all to the same file if you so desire.

Option five is used to create your mailing label output. You only have one format to create here.

Option six allows you to set special printer codes to change the style and print size of your reports without leaving the PRBASE environment.

Option seven lets you select your drive for your data disk, and change your output from printer to disk, etc. .

Option eight exits to the TI title screen.

USING YOUR DATA BASE:

After you finish designing your data input screen, then you can go to work entering your data. This you do by selecting item #2 from the main PRBASE menu. Data input is fast, and the sort and print routines are the quickest that I have ever seen.

To sort a data disk of 350 records takes approximately 35 seconds. Data may be retrieved with three search methods: Global disk search, Field disk search, and the super fast memory Index search. The memory Index search will find the data requested in under 1 second! Data found with these search methods may be output to your printer or disk file as screen dumps, tabular reports, or one- or two-across mailing labels, and may be incorporated into other documents with use of any Display/Variable 80 text editor. Data may be sorted by any field in seconds with the S command. You may also use the new selective indexing feature which allows you to sort on two different fields and therefore quickly isolate portions of your data base.

You have 18 single key commands and 11 double key commands available to help you in manipulating your data. If you think that this is a lot to remember, never fear. One of the single key commands is (H)elp which displays all of the commands except for 4 or the double key commands. This screen is instantly available and just as instantly removes itself with a FCTN 9 command.

There is a set of PRBASE UTILITIES written by John A. Johnson. They contain a menu with the following options: Copy Database Header (sectors 1-9), Copy a Group of Records, Copy a Single Record, Search and Select Records, Sort and Rewrite to Copy, Configure Drives, And Exit Program. Most are self-explanatory and are covered in the docs that come with the program. These utilities and docs are now included on the PRBASE disk.

PERSONAL THOUGHTS:

I admit to a certain amount of frustration when I first tried to use PRBASE. The discussion of Report Vector Tables and Report Format Design screens were a little confusing on the first try. After I set up a personal data base for home use, I then went back and set up a data base for the User Group mailing list. I finished designing the input screen in 10-15 minutes . From there I basically let the program design the tabular report, just deleting what I did not want on the final printout. You need to give some consideration to how you want your tabular printout to look when you design your data input screen. This will make the the process much simpler.

Overall, I feel that PRBASE is an excellent program, that is fairly easy to setup. After creating your data base the program is super fast and extremely easy and pleasing to use. The only negative thing I can think of, is that it would be nice to have a header at the beginning of each page instead of just at the first page.

All in all, this is well worth (many times over) the $10.00 that the author is asking you to send for his work. Please send the $10.00 to:

William Warren
2373 Ironton St.
Aurora, CO 80010

Or if you wish, you may pay the money to the PUNN treasurer, and he will send one check to Mr. Warren for the total of all received along with your name and address so that Mr. Warren can put you on his address list to be notified in case of future updates.

PUNN PAPER DRIVE NETS BIG BUCKS

BY AL KINNEY

Our paper drive last month netted the User Group Treasury over $25.00. Not bad, considering the low response to the drive. Seems like this is a good way for us to accomplish two things at once: First, we get the garage emptied out and Second, we benefit the Group. Maybe next time there will be a larger turnout and we will make even more. Special Thanks to Jim Smith for volunteering the use of his van. Against all my predictions, he was able to carry all the paper we collected.

BASIC BASICS
WHAT'S IN A STRING?

By Tom Nellis, Northcoast 99ers

How many times have you seen a statement similar to this.

```
10 A$=" THIS IS A TEST"
```

The value of A$ is now equal to whatever is in between the quotes. A$ will remain unchanged unless the program or user of the program alters its value.

The version of Basic TI has provided with the 99/4A console has 8 commands that allow you to manipulate strings in a variety of ways. These commands are ASC, CHR$, LEN, POS, RPT$, SEG$, STR$, and VAL. Some other versions of Basic offer a few more commands, but these 8 will do the job nicely.

The first four commands that we will examine are ASC, LEN, POS and VAL. What you should notice about these four commands is that the perform some type of function and return a numeric value. This is the reason their are no "$" trailing these commands.

A simple review of these commands follow:

ASC Returns the ASCII character code for the first character in the string.

```
A$="THIS IS A TEST"
A=ASC(A$)
```

"A" will now equal 84. The ASCII value of the letter "T". "T" is the first character in the string A$.

LEN Returns the total number of characters in the string.

```
A$="THIS IS A TEST"
A=LEN(A$)
```

"A" will now equal 14. The total number of characters in the string "A$".

POS Returns the position of the string you are looking for.

```
A$="THIS IS A TEST"
A=POS(A$,"IS",1)
```

"A" will equal 3. This number is the starting place of the first occurrence of "IS".

```
A=POS(A$,"IS",4)
```

"A" will now be equal to 6. The number 4 in the above example indicates the starting position in the string where the computer will be looking for "IS".

VAL This is used to convert a string value to a numeric value.

```
A$="12345"
A=VAL(A$)
```

"A" is now equal to 12345. You can now perform calculations on "A".

The next four commands return a string value. This is the reason for the "$".

CHR$ This is used to convert a number to an ASCII character.

```
X=65
B$=CHR$(X)
PRINT B$
```

B$ now equals "A". The ASCII value of the letter "A" is 65.

RPT$ This command must have a use but I haven't had to use it yet.

```
B$="TESTING "
C$=RPT$(B$,3)
PRINT C$
```

"TESTING TESTING TESTING " should be the answer. The number 3 indicates how many repetitions of the B$ you would like.

SEG$ This command is used to disect a string. You specify the string "A$", the position in the string you want to start at "6" and how many characters you want "2".

```
A$="THIS IS A TEST"
B$=SEG$(A$,6,2)
PRINT B$
```

"IS" is the answer. The word "IS" starts in the 6th position and is 2 characters long.

STR$ This command is the opposite of the VAL command. It will convert a numeric constant or variable to a string value.

```
X=65
A$=STR$(X)
PRINT A$
```

"65" is the answer. You cannot perform any type of arithmatic on it.

I hope that I have shed a little light on these commands when it comes to using functions. In the next article I will show you how these commands are used in a program.

TIPS FROM THE TIGERCUB

#39

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.

Over 13Ø original programs in Basic and Extended Basic, available on casette or disk, only $3.ØØ each plus $1.5Ø per order for PPM. Entertainment, education, programmer's utilities.
Descriptive catalog $1.ØØ, deductable from your first order.
Tips from The Tigercub, a full disk containing the complete contents of this newsletter Nos. 1 through 14, 5Ø original programs and files, just $15 postpaid.
Tips from the Tigercub Vol. 2, another diskfull, complete contents of Nos. 15 through 24, over 6Ø files and programs, also just $15 postpaid.

```
*****************************
*                           *
*  Tips from the Tigercub   *
*  Vol. 3 is now ready.     *
*  Another 62 programs,     *
*  routines, tips, tricks.  *
*  from Nos. 25 thru 32.    *
*  Also $15 postpaid. Any   *
*  two Tips disks $27 or    *
*  all 3 for $35 postpaid.  *
*                           *
*****************************
```

Nuts & Bolts (No. 1), a full disk of 1ØØ Extended Basic utility subprograms in merge format, ready to merge into your own programs. Plus the Tigercub Menuloader, a tutorial on using subprograms, and 5 pages of documentation with an example of the use of each subprogram. All for just $19.95 postpaid.
Nuts & Bolts No. 2, another full disk of 1Ø8 utility subprograms in merge format, all new and fully compatible with the last, and with 1Ø pages of documentation and examples. Also $19.95 postpaid, or both Nuts Bolts disks for $37 postpaid.
Tigercub Full Disk Collections, just $12 postpaid! Each of these contains either 5 or 6 of my regular $3 catalog programs, and the remaining disk space has been filled with some of the best public domain programs of the same category. I am NOT selling public domain programs - my own programs on these disks are greatly discounted from their usual price, and the public domain is a FREE bonus!
TIGERCUB'S BEST, PROGRAM-TUTOR, PROGRAMMER'S UTILITIES, BRAIN GAMES, BRAIN TEASERS, BRAIN BUSTERS!, MANEUVERING GAMES, ACTION REFLEX AND CONCENTRATION, TWO-PLAYER GAMES, KID'S GAMES, MORE GAMES, WORD GAMES, ELEMENTARY MATH, MIDDLE/HIGH SCHOOL MATH, VOCABULARY AND READING, MUSICAL EDUCATION, KALEIDOSCOPES AND DISPLAYS

For descriptions of these send a dollar for my catalog!

Answer to last month's challenge - for the longest possible one-liner, run the following "program to write a program" -

```
1ØØ OPEN #1:"DSK1.LONG",VARI
ABLE 163,OUTPUT
11Ø FOR J=1 TO 79 :: M$=M$&C
HR$(149)&CHR$(13Ø):: NEXT J
:: M$=CHR$(254)&CHR$(254)&M$
&CHR$(149)&CHR$(Ø):: PRINT #
1:M$ :: PRINT #1:CHR$(255)&C
HR$(255):: CLOSE #1
```

Then enter NEW, then MERGE DSK1.LONG, then LIST - over 34 lines long! But that one doesn't do anything, so try this -

```
1ØØ OPEN #1:"DSK1.LONG",VARI
ABLE 163,OUTPUT
11Ø FOR J=1 TO 52 :: M$=M$&C
HR$(162)&"X"&CHR$(13Ø):: NEX
T J :: M$=CHR$(254)&CHR$(254
)&M$&CHR$(162)&"X"&CHR$(Ø)::
 PRINT #1:M$ :: PRINT #1:CHR
$(255)&CHR$(255):: CLOSE #1
```

Again enter NEW, and MERGE DSK1.LONG, then RUN. You'll get a message BREAKPOINT IN 3251Ø (don't ask me why! Can anyone tell me?) but just enter RUN again. Then LIST it - over 24 lines long!

Explanation? Programs are saved in token code similar to MERGE format code. The maximum length of a record is 163 bytes - which is why MERGE files are D/V 163. The token for RANDOMIZE is ASCII 149, for the double colon is 13Ø. Repeating that 79 times takes only 158 bytes, plus one more RANDOMIZE, the two-byte tokenized line number and the mandatory ASCII Ø to end the record, totals 162.

Here's a spooky one for Hallowe'en -

```
1ØØ CALL CLEAR :: CALL MAGNI
FY(4):: CALL SCREEN(2) ! The
Blob by Jim Peterson
11Ø CALL CHAR(96,RPT$("3C7EF
FFFFFFF7E3C",4)):: J=-1
12Ø FOR L=1 TO 28 :: CALL SP
RITE(#L,96,16,L*4+2Ø,1Ø,Ø,L+
8):: NEXT L
13Ø FOR L=1 TO 28 :: CALL MO
TION(#L,Ø,L*J):: NEXT L
14Ø J=J*-1 :: GOTO 13Ø
```

Wes Johnston published an unusual sprite 2-liner in the Charleston Area 99ers newsletter. It is based on a CALL LOAD which freezes all sprite motion until they are turned loose by another CALL LOAD -

```
1ØØ R=PI*2/28 :: CALL CLEAR
:: CALL SCREEN(2):: CALL INI
T :: CALL LOAD(-318Ø6,96)::
FOR I=1 TO 28 :: CALL SPRITE
(#I,46,16,96,128,COS(I*R)*1Ø
,SIN(I*R)*1Ø):: NEXT I
11Ø CALL LOAD(-318Ø6,Ø):: GO
TO 11Ø
```

You might like to try adding my "jewels" to that -

```
1ØØ FOR CH=33 TO 6Ø :: FOR A
=1 TO 4 :: X=INT(8*RND+1)::
T$=SEG$("18243C425A667E81",X
*2-1,2):: A$=A$&T$ :: B$=T$&
B$ :: NEXT A :: CALL CHAR(CH
,A$&B$):: A$,B$="" :: NEXT C
H
11Ø R=PI*2/28 :: CALL CLEAR
:: CALL SCREEN(2):: CALL INI
T :: CALL LOAD(-31BØ6,96)::
FOR I=1 TO 28 :: CALL SPRITE
(#I,32+I,INT(14*RND+3),96,12
8,COS(I*R)*1Ø,SIN(I*R)*1Ø)::
 NEXT I
12Ø CALL LOAD(-318Ø6,Ø):: GO
TO 12Ø
```

Also try CALL MAGNIFY(2)

And, here is a companion program to the TAKE AWAY in Tips #35 -

```
1ØØ CALL CLEAR :: CALL TITLE
(5,"ADD & CARRY")!by Jim Pet
erson
11Ø DISPLAY AT(3,1Ø):"COPYRI
GHT":TAB(1Ø);"TIGERCUB SOFTW
ARE":TAB(1Ø);"FOR FREE":TAB(
1Ø);"DISTRIBUTION":TAB(11);"
SALE PROHIBITED"
12Ø CALL PEEK(-28672,A@):: I
F A@=Ø THEN 16Ø
13Ø DATA FINE,NO,GOOD,UHOH,R
IGHT,TRY AGAIN,YES,THAT IS N
OT RIGHT
14Ø FOR J=1 TO 4 :: READ RIG
HT$(J),WRONG$(J):: NEXT J
15Ø FOR D=1 TO 1ØØØ :: NEXT
D :: CALL DELSPRITE(ALL)
16Ø CALL CLEAR :: CALL CHAR(
95,"FFFF"):: CALL MAGNIFY(2)
:: RANDOMIZE :: CALL SCREEN(
14):: FOR SET=5 TO 8 :: CALL
 COLOR(SET,16,1):: NEXT SET
17Ø CALL CHAR(12Ø,"E7ØØ42ØØ1
8ØØ7EØØØØE7ØØ42ØØ99423CE7ØØ4
2ØØ99423CØØE7ØØ4218ØØ3C42ØØ"
)
18Ø CALL CHAR(124,"ØEØØØ4Ø1Ø
ØØ7ØBØØ7ØØØ2ØBØØØEØ1ØØØØ")
19Ø DISPLAY AT(3,8):"ADD AND
 CARRY" :: CALL CHAMELEON
2ØØ CALL COLOR(14,2,2):: CAL
L HCHAR(4,4,143,2):: CALL HC
HAR(5,4,143,2):: CALL SPRITE
(#25,12Ø,11,25,25)
```

```
210 T=T+1 :: IF T=6 THEN T=0
 :: GOTO 250
220 Z=INT(8*RND+2):: IF Z=Z2
 THEN 220 ELSE Z2=Z
230 Y=INT(Z*RND):: IF Y=Y2 T
HEN 230 ELSE Y2=Y :: X=Z-Y
240 N=1 :: GOSUB 470 :: GOTO
 210
250 T=T+1 :: IF T=11 THEN T=
0 :: GOTO 290
260 X=INT(10*RND):: IF X=X2
THEN 260 ELSE X2=X
270 Y=INT(10*RND):: IF Y=Y2
OR X+Y<10 THEN 260 ELSE Y2=Y
 :: Z=X+Y
280 N=1 :: GOSUB 470 :: GOTO
 250
290 T=T+1 :: IF T=11 THEN T=
0 :: GOTO 330
300 X=INT(90*RND+10):: IF X=
X2 THEN 300 ELSE X2=X
310 Y=INT(90*RND+10):: IF Y=
Y2 THEN 310 ELSE Y2=Y :: Z=X
+Y
320 N=2 :: GOSUB 470 :: GOTO
 290
330 X=INT(900*RND+100):: IF
X=X2 THEN 330 ELSE X2=X
340 Y=INT(900*RND+100):: IF
Y=Y2 THEN 340 ELSE Y2=Y :: Z
=X+Y
350 N=3 :: GOSUB 470 :: GOTO
 330
360 R=96 :: CC=96 :: FOR J=1
 TO N :: CALL SPRITE(#J,48+A
(J),11,R,CC):: CC=CC+16 :: N
EXT J
370 R=116 :: CC=96 :: FOR J=
1 TO N :: CALL SPRITE(#4+J,4
8+B(J),11,R,CC):: CC=CC+16 :
: NEXT J
380 CALL HCHAR(18,12,95,N*3)
:: CC=CC-16 :: CALL SPRITE(#
22,43,16,R,80):: RETURN
390 R=140 :: FOR J=LEN(STR$(
Z))TO 1 STEP -1 :: CALL SPRI
TE(#20,63,11,R,CC)
400 CALL KEY(3,K,ST):: IF ST
<1 OR K<48 OR K>57 THEN CALL
 PATTERN(#20,32):: CALL PATT
ERN(#20,63):: GOTO 400
410 CALL DELSPRITE(#20):: CA
LL SPRITE(#12+J,K,11,R,CC)
420 IF K-48<>C(J)THEN GOSUB
480 :: CALL DELSPRITE(#12+J)
:: CALL SPRITE(#20,63,11,R,C
C):: GOTO 400
430 IF A(J-W)+B(J-W)>9 THEN
CALL SPRITE(#28,49,16,80,CC-
16)
440 CC=CC-16 :: NEXT J :: GO
SUB 510 :: RETURN
450 FOR J=1 TO LEN(STR$(X)):
: :: A(J)=VAL(SEG$(STR$(X),J
,1)):: NEXT J :: FOR J=1 TO
LEN(STR$(Y)):: B(J)=VAL(SEG$
(STR$(Y),J,1)):: NEXT J
460 FOR J=1 TO LEN(STR$(Z)):
: C(J)=VAL(SEG$(STR$(Z),J,1)
):: NEXT J :: W=LEN(STR$(Z))
-LEN(STR$(X)):: RETURN
470 GOSUB 450 :: GOSUB 360 :
: GOSUB 390 :: FOR D=1 TO 20
0 :: NEXT D :: CALL DELSPRIT
E(ALL):: DISPLAY AT(18,1)::
CALL CHAMELEON :: CALL SPRIT
E(#25,120,11,25,25):: RETURN
480 DATA 123,124,125,123,124
,125,123,120
490 IF A@=0 THEN 500 :: CALL
 SAY(WRONG$(INT(4*RND+1)))
500 RESTORE 480 :: FOR JJ=1
TO 8 :: READ P :: CALL PATTE
RN(#25,P):: XX=2^250 :: NEXT
 JJ :: RETURN
510 DATA 121,122,121,122,121
,122
520 IF A@=0 THEN 530 :: CALL
 SAY(RIGHT$(INT(4*RND+1)))
530 RESTORE 510 :: FOR JJ=1
TO 6 :: READ P :: CALL PATTE
RN(#25,P):: XX=2^250 :: NEXT
 JJ :: RETURN
540 SUB CHAMELEON
550 M$="1800665AC342DB667E18
8100995AC3A5E78142BD24DB6600
81429924007E5AC3A53C241B00FF
DB5AFF7EFF0099188100660018"
560 RANDOMIZE :: CALL CHAR(1
28,SEG$(M$,INT(43*RND+1)*2-1
,16)):: X=INT(14*RND+3)
570 Y=INT(14*RND+3):: IF Y=X
 THEN 570 :: CALL COLOR(13,X
,Y)
580 CALL HCHAR(1,2,128,30)::
 CALL HCHAR(24,2,128,30):: C
ALL VCHAR(1,31,128,96):: SUB
END
590 SUB CHAMWIPE
600 T=T+1+(T=2)*2 :: ON T GO
TO 610,620
610 CALL VCHAR(1,3,128,768):
: GOTO 630
620 CALL HCHAR(1,1,128,768)
630 CALL CLEAR :: SUBEND
640 SUB TITLE(S,T$)
650 CALL SCREEN(S):: L=LEN(T
$):: CALL MAGNIFY(2)
660 FOR J=1 TO L :: CALL SPR
ITE(#J,ASC(SEG$(T$,J,1)),J+1
-(J+1=S)+(J+1=S+13)+(J>14)*1
3,J*(170/L),10+J*(200/L))::
NEXT J
670 SUBEND
```

A mathematical curiosity -

```
100 !MAGIC NINES by Jim Pete
rson
110 CALL CLEAR
120 INPUT "TYPE ANY 3-DIGIT
NUMBER OF 3 DIFFERENT DIGITS
 ":N :: IF N<>INT(N)OR N>999
 OR N<0 THEN 120
130 N$=STR$(N):: IF N<100 TH
EN N$="0"&N$
140 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 ">>>THREE DIF
FERENT DIGITS<<," :: GOTO 12
0
150 PRINT :: N2$="" :: FOR J
=1 TO 3 :: N2$=SEG$(N$,J,1)&
N2$ :: NEXT J :: N2=VAL(N2$)
:: D=ABS(N-N2)
160 PRINT N$;" BACKWARDS IS
";N2$: :
170 N3=ABS(N-N2):: N3$=STR$(
N3):: IF N3<100 THEN N3$="0"
&N3$
180 IF N>N2 THEN PRINT N$;"
MINUS ";N2$;" EQUALS ";N3$:
:ELSE PRINT N2$;" MINUS ";N$
;" EQUALS ";N3$: :
190 FOR J=1 TO 3 :: N4$=SEG$
(N3$,J,1)&N4$ :: NEXT J
200 PRINT N3$;" BACKWARDS IS
 ";N4$: :N3$;" PLUS ";N4$;"
IS 1089": :"I KNEW THAT WOUL
D BE THE": :"ANSWER!": :"LIS
T THE PROGRAM AND SEE!"
210 !!!!!!!!!!!!!!!!!!!!!!!!!
220 !  THE ANSWER WILL BE  !
230 !         1089         !
240 !!!!!!!!!!!!!!!!!!!!!!!!!
```

```
100 DISPLAY AT(8,10)ERASE AL
L:"SHENANDOAH": : :"  Across
 the wide Missouri": : : : :
 : : : : : : :"programmed by
 Jim Peterson"
110 FOR D=1 TO 1000 :: NEXT
D :: CALL CLEAR :: DIM S(24)
:: RANDOMIZE :: M$="4218005A
007E9981005A24DBC31824243C5A
7EA56618003CDB66BD3CA542187E
5AC324425A18A51866810081187E
423CBDDBC3" :: R=1
120 FOR CH=40 TO 136 STEP 8
130 CALL CHAR(CH,SEG$(M$,INT
(43*RND+1)*2-1,16)):: CALL H
CHAR(R,1,CH,64):: R=R+2*ABS(
R<23)
140 NEXT CH :: R=0 :: FOR SE
T=2 TO 14 :: X=INT(14*RND+2)
150 Y=INT(14*RND+2):: IF Y=X
 THEN 150
160 CALL COLOR(SET,X,Y)
170 NEXT SET :: CALL CLEAR :
: CALL COLOR(1,5,5):: CALL V
CHAR(1,29,1,192):: CALL SCRE
EN(16):: F=262 :: FOR N=0 TO
 23 :: S(N)=INT(F*1.0594630
4^N):: CALL SOUND(-999,S(N),
0)
180 NEXT N
190 DATA 2,1,1,1,6,1,1,1,6,2
,6,1,1,1,6,1,8,8,1,10,10,1,1
1,11,1,15,6,3,13,6,2,13,11
200 DATA 1,18,10,1,17,17,4,1
5,11,1,11,15,1,13,13,1,15,11
,1,13,13,1,10,10,3,13,10
210 DATA 2,13,13,2,13,10,1,1
5,10,1,10,15,2,15,15,1,15,10
,1,10,10,1,13,13,1,10,10
220 DATA 1,8,3,3,6,3,2,6,6,2
,8,8,4,10,1,1,10,6,1,6,6,1,1
0,10,1,15,15
230 DATA 2,13,1,2,13,5,2,13,
10
240 DATA 1,6,6,1,8,8,6,10,6,
2,3,3,2,8,5,1,8,1,3,6,1,7,6,
1
250 A=1 :: B=1 :: E=5
260 FOR J=1 TO 144 STEP 3 ::
 CALL HCHAR(A,E,32,T*4):: CA
LL HCHAR(A+1,E,32,T*4):: CAL
L HCHAR(B,E,32,T*4):: CALL H
CHAR(B+1,E,32,T*4):: READ T,
A,B :: E=17-T*2
270 CALL HCHAR(A,E,32+INT((A
+1)/2)*8,T*4):: CALL HCHAR(A
+1,E,32+INT((A+1)/2)*8,T*4):
: CALL HCHAR(B,E,32+INT((B+1
)/2)*8,T*4)
280 CALL HCHAR(B+1,E,32+INT(
(B+1)/2)*8,T*4):: FOR D=1 TO
 T :: CALL SOUND(-999,S(A),0
,S(B),7)
290 NEXT D
300 NEXT J :: LL=0 :: FOR SE
T=2 TO 14 :: X=INT(15*RND+2)
310 Y=INT(15*RND+2):: IF Y=X
 THEN 310
320 CALL COLOR(SET,X,Y):: CA
LL SOUND(-999,S(6),LL,S(1),L
L):: LL=LL+2
330 NEXT SET :: RESTORE :: G
OTO 260
>>>>>>>>MEMORY FULL<<<<<<<<
```

PRODUCT REVIEW: Pre-Scan IT!

by Duane Goodman

This fine program is written by J. PETER HODDIE and is distributed by ASGARD SOFTWARE.

What this program does is take an existing Ex-Basic program, re-writes it in pre-scan format, and inserts the appropriate coding to turn on and off the Pre-Scan feature when necessary. It also will replace up to 5 numbers in the program with the variables """, "[", "]", "_", and "\". The reason for this is that it takes only 1/3 the space to store this type of variable as opposed to the number itself. Pre-Scan IT will also remove all of the "REM" comments in the program and text and replace them with exclamation points, thus saving a lot of program space.

Using Pre-Scan IT!

The first thing you must do is RESequence the program you wish to Pre-Scan. Pre-Scan IT needs the area from 1 to 100 for its own use. After you have RESequenced your program, then save it in the following format:

```
SAVE DSKn.XXXXXXXXXX,MERGE
```

Where XXXXXXXXXX represents a filename different from the original program. After saving the program in MERGE format, insert the PSI disk and type the following to load and execute Pre-Scan IT:

```
"RUN DSK1.LOAD"
```

After running your program thru Pre-Scan IT, it will again be written back to the disk in MERGE format. To return it to PROGRAM format you must type in:

```
OLD DSKn.XXXXXXXXXX
```

Then after is is loaded into memory, you would type in:

```
SAVE DSKn.XXXXXXXXXX
```

and your file will be written back to the disk in program format ready to use.

I used Pre-Scan IT on an AUTO MAINTENANCE program that I purchased that was slow. The initial loading time for the program was 30 seconds . After running it thru Pre-Scan IT, it loads in 18 seconds . That is a 40% increase in speed of loading. There is also the same speed increase going from one menu selection to another within the program.

If you either use a lot of Ex-Basic programs or program in Ex-Basic, this program is well worth the $10.00 asking price.

To order send $10.00 to:

ASGARD Software
P.O. Box 10306
Rockville MD 20850

For more information or to order by credit card call 301-345-2492

NEWSLETTER EDITOR NEEDED

BY AL KINNEY

If you have ever had the desire to be a Newsletter Editor, now is your opportunity! All you need to do is contact one of the Officers at the next meeting, or come to the Board meeting.

I will be resigning the post, effective after the completion of this issue, to pursue other interests. I intend to stay involved with PUNN, and will continue to work on the BBS, but have several other projects that have been waiting for a long time. Thanks for all of your support.

Al Kinney

MISSING SOMETHING???

BY RON MAYER

If you were over to my home about a month ago, and left a brown/green plaid jacket, please claim it at the next meeting. Thanks.

P.O.BOX 15Ø37
EAST PORTLAND STATION
PORTLAND, OR. 97215

ALL GENERAL MEETINGS ARE HELD
ON THE FIRST TUESDAY OF EACH
MONTH. AT THE PGE BLDG.
3700S. E. 17TH. PORTLAND. OR

!! NEXT MEETING DATE !!

NOVEMBER 4th
ELECTION DAY
1986