WORDPLAY

The PUNN Newsletter - Portland, Oregon
November 1988

Club Officers

President   Al Kinney 640-5860
Vice-President   Dale Kirkwood 645-4117
Secretary   Don Barker 223-1749
Treasurer   Chuck Neal 642-7292

PUNN Staff

Librarians   Ron Mayer 232-7363
                 Walt Morey 239-5105
                 Jim Thomas 284-2425
Hardcopy   Mike Calkins 636-1839
Program Chairman Ted Peterson 244-1587
Workshop Chairman   Jim Smith 233-0273
Membership Chairman Terry Priest 649-9583
Newsletter Editor   Charles Ball 639-0466
16576 SW Matador Lane-King City, OR 97224

BBS Committee

Chairman: Al Kinney 640-5860
          Ron Mayer 232-7363
          Mike King 357-4413

BBS Phone Number   503/233-6804

From The President

As we near the end of the year, a couple of very important events are approaching. First, our Annual Officer's election will be held in December. The Nominating Commitee has been at work, helping to put together a good slate for you to vote on at the election.

The other thing coming up this fall is something we set out as one of our goals at the first of the year. The goal was to get PUNN involved in a community service action. To this end, I am announcing the start of the First Annual PUNN Canned Food Drive, beginning with the next general meeting. We have about 100 paid members in PUNN, so it seems to me that we should be able to set as a goal, the collection of at least 150 cans (or pounds) of food for the needy.

Now, obviously, not everyone can come to every meeting, so it is important for those of us who are there to bring at least two cans of food to each of the November and December meetings. But, it would also be great to see EVERYONE come to at least one of the meetings. Let's work to make this a success, and I'm sure we'll all feel richer for it.

We'll also be collecting newspapers and donating the proceeds of that effort to the needy. Be sure to help where you can.

Al Kinney

News and Views

This month we have some good news and some bad news- - -Mike King got married (see elsewhere in this issue), but we are saddened by the passing of member Peter Rawlinson-our expression of sympathy goes out to his family- - -The Library is going to be revitalized-On Saturday morning, October 29th a group is going to meet for the express purpose of cataloging the disks and updating the procedures for getting programs to the members-If you can help with this activity speak to Al Kinney-Your help is needed- - -Plan to come to the November meeting and bring a couple of cans of food to be donated to a worthy charity-Let's show Portland that PUNN is a good neighbor that participates in community affairs- - -Don't throw away those old newspapers!-We're going to have another paper drive soon-This time we're going to donate the procedes to charity- - -Volunteers are always welcome to help the Editor with the newsletters-He has had help recently from Ashley Reed, Paul Heerman, John Usher, Bill McCabe, Jim Klausmeier and Don Steffan-You too can help make the newsletter one of the best-With many helping out in this manner the bulletin has a much wider appeal- - -The Nominating Committee will report its slate of officers for 1989 at the November meeting-Nominations from the floor are open-If you or someone you know would desire to serve, let your wishes be known- - -Chuck Neal reports a bank balance of $1571.05 and this is after the purchase of the 2400 baud modem for the BBS- - -Remember you are welcome at the Board Meetings-The next one will be held at the home of Chuck Ball- - -Ted Peterson, Program Chairman says we'll have TI-Writer and Multiplan programs in January and February-He wants to hear from you as to the type of program you want.   ccb, editor

Name That Phone

(This program was originally written by Ed Machonis of the QB-99er's, Bayside NY, but has been edited and improved by our own Ashley Reed-Thanks Ashley for a nice job!-ed.

Do you remember calling TI-CARES? Never had to look up that number, it always was at the tip of your fingertips! And what an asset it was to TI. I don't believe that helpline cost TI a penny. It paid its weight in free advertising. Knowing that help was at the other end of a toll free line sold many a computer.

Today, many businesses strive for a telephone number that can be easily remembered. Perhaps there is a word or phrase hidden in your own phone number. But how to discover it? You could look over the dial and see what letters are involved and try arranging them into words. Good luck!

Each of the seven digits in your phone number, excluding ones and zeros, can represent any one of three letters of the alphabet. The number of possible combinations is 3 to the 7th. power, or 2187. Try arranging them into words some rainy weekend.

Sounds like one of the tasks we bought our computers for and it is. The program listed here can do the job in just over 11 minutes. It will present you with every combination of letters existing in your number. It will display them on screen or send them to your printer. The printout, in 6 columns takes up about six pages. Not bad for a screenful of code.

When you RUN, the program it will prompt you to enter your phone number, one digit at a time. It will then ask you to choose Screen or Printer. (You may enter any number from 1 to 255 for the Printer.)

There are no letters assigned to One or Zero. If you enter these numbers into the program, it will display asterisks for zeros and number signs for ones.

The screen display is in two columns and scrolls by just about fast enough for you to follow. Should you spot some interesting combinations you would like a closer look at just break the program with FCTN 4. When you

are ready to continue, just tell the computer to do so by typing CON and it will resume where it left off.

If you don't find a word, you should certainly be able to find a phrase, the initial letters of which would represent your number. Try to work out a phrase which is easily associated with you.

In my own case, the best I could find was EGDRAMA which I turned into the mnemonic "Ed's Great DRAMA." Using initial letters, I can also be reached with "Ed Is Forever Programming A Marvelous Computer." Avoid using numbers in your mnemonics, like "Ed Has Damaged Seven Brand New Cars." Your friends start to wonder, "Let's see, is it SIX or SEVEN new cars that maniac has destroyed?" And then there are the characters who will persist in dialing the number 7 instead of the initial letter "S"

There may be a fantastic mnemonic hiding in your phone number, but you won't know it unless you run this program. Good Luck!

```
1 !*****NAME THAT PHONE******* 10 FOR J=1 TO 3 :: G$=G$&SEG
* No 1's or 0)'s Please * $(E$,6+J,1)
*        A TIny Gram      * 11 FOR K=1 TO 3 :: G$=G$&SEG
*      by Ed Machonis     * $(E$,9+K,1)
***QB-99'ers Bayside NY** 12 FOR L=1 TO 3
2 !*******REVISED BY******** 13 G$=G$&SEG$(E$,12+L,1):: F
3 !*******ASHLEY REED******* OR M=1 TO 3 :: G$=G$&SEG$(E$
4 !**PORTIAND USER'S OF 99** ,15+M,1):: FOR N=1 TO 3
5 DEF S$=.::$(G$,1,LEN(G$)-1 14 G$=G$&SEG$(E$,18+N,1):: P
)::A$="*****&^^^FGHIJKLMN RINT #P:G$,:: G$=S$ :: NEXT
OPRSTUVWXY" :: !... C=1 TO 7 N :: G$=S$ :: NEXT M :: G$=S
6 INPUT "ENTER DIGIT "&STR$( $ :: NEXT L :: G$=S$ :: NEXT
C)&" OF PHONE # ":D :: E$=E$ K :: G$=S$ :: NEXT J :: G$=
&SEG$(A$,D*3+1,3):: NEXT C   S$ :: NEXT H :: NEXT F
7 PRINT :"0=SCREEN":"1=PRINT 15 INPUT "CHANGE DISK THEN P
ER":"CHOICE (0/1)",:: INPUT RESS ENTER":KY$
P :: IF P THEN OPEN #P:"PIO" 16 RUN "DSK1.LOAD"
8 FOR F=1 TO 3 :: G$=""&SEG$ 17 !COMMENTS.
(E$,F,1)                     18 !IF YOU HAVE A 1 OR 0 IN
9 FOR H=1 TO 3 :: G$=G$&SEG$ YOUR P  NUMBER YOU MAY WI
(E$,3+H,1)                   SH TO CHANGE LINE 2 TO ABCDE
                             F INSTEAD OF ******
```

Wedding Bells

Wedding Bells have rung once again in PUNN Land. Ex-Treasurer Mike King and Mary Ann (Mele) King were married on Saturday October 8. The wedding took place at 4:30pm at Holy Reedemer Church in North Portland.

They left Sunday for a one week vacation in Hawaii. The bride's immediate family and friends attended as did Mike's family who came from New York.

All of us in PUNN offer our congratulations for a happy life.

Perhaps the bride is into computers and will come to the meetings.

```
*****************************************
*                                       *
*    Murphy's Rule:                     *
*                                       *
*    When all is said and done          *
*         more will be said             *
*            THAN done.                 *
*                                       *
*                                       *
*****************************************
```

Hard Copy

Mike Calkins, our Hard Copy Librarian, asks once again for the return of outstanding books.

These books are for all to use and enjoy, but they are of no value if they are not returned and made available for other members.

Mike will be on vacation at the November meeting and he asks that you return the books to Jim Thomas during the meeting.

Canned Food

Your are requested to bring a can or two of food when you attend both the November and December meetings.

We're going to donate these gifts to a worthy charity and this is a good time to get into the 'Christmas Spirt'.

==

Slot Machine

Planning a trip to Reno or Las Vegas? If so you will want to try this program and practice up for your trip.

You've probably been there in the past at one time or another. So you will recall the the whirring and the sounds of the slot machines and found yourself wondering what the fascination was.

You"ll soon find out when this program turns your computer into a fabulous Nevada style fruit machine. All the playing instructions you need will appear on the screen. At certain points you will be presented with a list of options.

When you see:

Insert, Hold, Play OR End

enter your choice by typing the first letter of the option you want, for instance P keeps your machine playing. The reels are numbered 1, 2, and 3. If you want to hold one or more reels, type in the appropriate number or numbers after you have entered H for Hold.

Lights will flash and music play as the wheels whiz around. Have you won this time?

Keep playing and sooner or later you are bound to hit the Jackpot.

```
10 REM TI(EXTENDED)-SLOT MAC
HINE
20 CALL CLEAR :: RANDOMIZE
30 ... WF((13),R(2,13),JAC(13
):: JA..(5)=-1 :: JAC(6)=-1 :
: ...   ")=-1
40 : .. VV=1 TO 13 :: READ WF
(VV),(k1),VV),R(2,VV):: NEXT
VV
50 DATA 110,0,0,116,0,0,115,
0,0,109,0,0,106,8,50,105,8,4
0,114,7,30
60 DATA 113,5,25,104,3,20,13
6,3,20,137,2,10,128,2,10,112
,0,0
70 DIM JPX(4),JPY(4),JF(4),J
CC(4):: FOR JO=1 TO 4 :: REA
D JPX(JO),JPY(JO),JF(JO):: N
EXT JD
80 DATA 11,8,110,12,8,116,14
,8,115,15,8,109
90 NJ$=CHR$(138)
100 DIM WPX(3),WPY(3),V(4),H
VV(3):: WPX(1)=11 :: WPX(2)=
13 :: WPX(3)=15 :: FOR I=1 T
O 3 :: WPY(I)=10 :: NEXT I
110 DIM HPX(3),HPY(3),HF$(3)
:: FOR I=1 TO 3 :: HPX(I)=WP
X(II):: HPY(I)=13 :: NEXT I
120 H1$=CHR$(107):: NH$=CHR$
(.. :: HH$=...('96):: HO$=H
1$... 1$... 1$...
130 ... 6PX(4),or1(4),6V(4):
: Gr..(.;=22 :: 6PY(1)=1
140 6PX(2)=27 :: 6PY(2)=2 ::
6PX(3)=22 :: 6PY(3)=3 :: 6P
X(4)=17 :: 6PY(4)=2
150 NJ=4 :: WH=0 :: NG=0 ::
6W=0 :: WI=0 :: WT=-1
160 GOTO 1740
170 REM BLINK S1$/S2$ AND GE
ET IN
180 DISPLAY AT(PLY,PLX):S1$;
:: CH=0 :: FOR DE=1 TO 10
190 IF CH=0 THEN CALL KEY(0,
6ET,CH1
200 NEXT DE :: DISPLAY AT(PL
Y,PLX):S2$;:: IF CH=1 THEN R
ETURN
210 FOR DF=1 TO 10 :: IF CH=
0 THEN     KEY(0,6ET,CHI
... NEXT DE :: IF CH=1 THEN
.. .RN
..:: GOTO 180
240 REM AND INCREMENT TO MON
EY
250 FOR AO=66WI(IN)TO IN STEP
SG%...
260 :    AY AT(1,7):HO+AD;
270 ir 5 %-:.%=-50 THEN 290
.. CALL .. %: -50,200,2)::
.. 300
... FOR SO=500 TO 700 STEP 1
00 :: CALL SOUND(-50,60,2)::
NEXT SO
300 NEXT AD :: HO=MO+IN :: R
ETURN
310 REM REMOVE DOUBLE
320 DO=0 :: CALL HCHAR(2,3,3
2,7)
330 FOR JO=1 TO 4 :: CALL HC
HAR(JPY(JO),JPX(JO),JF(JO)
):: JCC(JO)=0 :: NEXT JO
340 NJ=4 :: F...
350 CALL HCH... .4,3,32,28)::
RETURN
360 REM JACKPOT
370   TI=1 TO 4 :: CALL HC
HAR(4,3,32,JC)
380 FOR C=1 TO JC :: CALL SO
UND(-100,150+20*C,0):: DISPL
AY AT(4,C):"J";::: NEXT C
390 NEXT TI :: R" .. .
400 REM DEFINE C '-" .
410 CALL CHAR(96,":::...i..FF
FFFFFF")
420 CH$="0F0F0F0F0F0F0F0F0F" :
: CALL C .. ".."H$)
430 CALL .. .. .CH$)
440 CALL ...(Y,6,1,10,2,16
,11,7,16,12,16,6,13,11,16,14
,13,14)
450 ... CHAR(107,"FFBBBBBB
3BB...";
460 ... CHAR(108,"FFFFFFFF
FFFFFF")
470 A$="1898FF3D3C3CE404"
480 CALL CHAR(109,A$):: CALL
CHAR(... "#"
490 A$ ". ...BC3C3C2720"
500 CALL CHAR(110,A$):: CALL
CHAR(116,A$)
510 CALL CHAR(136,"02043C566
A566A3C")
520 CALL CHAR(114,"00006C7C7
C381000")
530 CALL CHAR(112,"0204387C7
C7C...."
540 .. CHAR(104,"02020C3C7
878.. :"
550 .. CHAR(128,"004060607
03C. :."
560 ... CHAR(137,"020413C37
C7C..."
570 CALL CHARPAT(36,A$):T CA
LL CHAR(106,A$)
580 CALL CHARPAT(63,A$):: CA
LL CHAR(120,A$)
590 CALL CHAR(113,"081C3E7F3
EE1C0800")
600 ... CHAR(104,"081C2A772
A08. :."
610 ... CHAR(105,"0010387C7
C10. .")
620 CALL CHAR(138,"000000000
000...."
630 ... -N
640 . "..DRAW SCREEN
650 CALL CLEAR :: CALL SCREE
N(12)
660 CALL HCHAR(7,12,96,7)
670 FOR I=8 TO 12 :: CALL HC
HAR(I,11,96,9):: NEXT I
680 CALL .. .. -.20,96)
690 CALL ....- .. .20,97,4)
700 CALL ... .20,96)
710 CALL  .. :3,12,96,7)
720 FOR I=14 TO 19 :: CALL H
CHAR(I,13,96,5):: NEXT I
730 CALL HCHAR(20,12,96,7)
740 ... .. ..11,96,9)
750 .. .. ... .10,96,11)
760 ...:..AT A(11,1):"TOTAL:0
770 DISPLAY AT(10,3):CHR$(11
2);" = 1 ;
780 FOR VV=12 TO 5 STEP -1
790 .. . AY AT(23-VV,1):"-";
RPT$"-..$(WF(VV)),2);"-=";R(
1,VV);
800 DISPLAY AT(23-VV,20):RPT
$(CHR$(WF(VV)),1,3);"="$,R(2,VV
);
810 IF JAC(VV)THEN DISPLAY A
T(23-VV,27):"J";
820 NEXT VV :: R" . .
830 REM AT ADAPT . "BLES
840 WT=WT+1 :: IF W.// THEN
HB=-1 :: WI=0
850 FOR HO=1 TO 3 :: HF$(HO)
=NH$ :: DISPLAY AT(HPY(HO),H
PX(HO)):NH$;:: NEXT HO
860 IF DO THEN 910
870 FOR JO=1 TO 4 :: IF JCC(
JO)=0 THEN 900
880 JCC(JO)=JCC(JO)-1 :: IF
JCC(JO)>0 THEN 900
890 NJ=NJ+1 :: CALL HCHAR(JP
Y(JD),JPX(JO)+2,JF(JO))
900 NEXT JO :: GOTO 920
910 DC=DC-1 :: IF DC=0 THEN
GOSUB 320
920 IF JA THEN DISPLAY AT(4,
...:" ";:: JC=JC-1 :: IF JC=
0 THEN JA=0
930 RETURN
940 REM ## NOT ENOUGH MONEY
950 HB=-1 :: GOSUB 320
960 JA=0 :: CALL HCHAR(4,3,3
2,15)
970 GOSUB 350 :: :  AY AT(
24,1):"IF  AT OF -. ";
980 S1$=":.% --T .;;/ :: S2$=
RPT$(" ",1v;:: PLX=1 :: PLY
=2 :: GOSUB 180
990 IF GET=ASC("P")THEN 980
ELSE - -N
1000 --- 11 HOLD POSSIBLE
1010 .. .AY AT(24,1):"INSER
T, HOLD,PLAY OR END";
:... S1$=HO$ :: S2$=HF$(1)&H
 .%: $(2)&HH$&HF$(3)
1030 PLX=11 :: PLY=13 :: GOS
UB 180
1040 IF GET(49 OR GET)51 THE
N RETURN ELSE HO=GET-48
1050 NH=NH+1 :: IF HF$(HO)=N
H$ THEN HF$(HO)=H1$ ELSE HF$
(HO)=NH$
1060 ...: .. 1020
1070 . - ## NO HOLD
1080 .  : 350 :: DISPLAY AT
(24,1):".% -:.PLAY OR END";
1090 S1$ -- :% CHR$(120),3)::
S2$=RPT$(CHR$(96),3):: PLX=
12 :: PLY=19 :: GOSUB 180
1100 RETURN
1110 --- ## WHAT TO DO WITH
WINN...
1120 . . : 350 :: IF NOT HB
THEN ::.. LAY AT(24,1):"HOLD,
.;
1130 DISPLAY AT(24,7):"GAMBL
E OR .. :CT";
1140 .. : 1090
1150 -.- ## SPIN GAMBLE WHEE
LS
1160 :::: FOR I=7 TO 10 :: CALL
C-.- ..20,32):: CALL HCHAR(I
,i,20,60)):: N- .. I
1170 .:%- 5" %" 4000,-7,29)
1180 - - I=10 iv 7 STEP -1
: C-. HCHAR(I,20,60):: CALL
%...-:,I,20,97):: NEXT I
1190 FOR WD=1 TO 3 :: IF HF$
.%: WH$ THEN : AY AT(WPY
%.%: WD)):: -# 138);
:.:. %: WD=1 TO 3 :: IF HF$
# .% H$ THEN 1240
... F1=INT(PV.1^^+1):: IF
.. THEN IF . :)()0 THEN
V(WD)=13 ELSE V(WD)=FI :: J
CC(F1)=-( ::   1240
..:% V(WO)=5- . ")-(F1)(0)-
.. :3)-(F1)23)-(F1)36)-(F1
49)-(F1)68)-(F1)87)
1240 FOR DF=2 TO 300 :: NEXT
DE :: .%  AY AT(WPY(WD),WP
X(WD)):.%- "  ;
1250 LA.. % - -100,300,2):
: CALL . - # - WD),WPX(WD)
+2,WF(1 .% .
1260 CALL SOUND(4000,-7,29):
: NEXT WD :: CALL SOUND(-1,-
2,30):: RETURN
1270 REM TAKE CARE OF JOKERS
1280 J=0 :: FOR WD=1 TO 3 ::
VV=V(WD)
1290 IF VV)4 THEN FV=VV :: G
OTO 1320
:". JW=WD :: J=J+1 :: IF JC
I .. )0 THEN 1320
.:::: DISPLAY AT(JPY(VV),JPX(
VV)):NJ$;:: JCC(VV)=20 :: NJ
=NJ+1
1320 NEXT WD :: DO=(NJ=0)::
IF DO THEN DC=15 :: DISPLAY
AT(2,1):"..  E!";
.::: RETURN
1340 REM ## COMPUTE #...'NGS
1350 HV=13 :: ON J+1 .. .13
60,1380,1360,1410
1360 IF NOT(JA AND FV)10)THE
N HV=FV
1370 ... 1410
1380 .  .V(3)=: V(4)=V(1)::
IF JA AND V(JW-1)()V(JW+1)T
HEN 1410
1390 V(0)=15 :: V(4)=15
1400 IF V(JW+1))V(JW-1)THEN
HV=V(JW-1)ELSE HV=V(JW+1)
1410 - - WD=1 TO 3 :: IF V(W
D)<5 . % HVV(WD)=HV ELSE HV
. . . I
1420 NEXT WD :: IF HVV(1)(H
VV(2)OR HVV(2)()HVV(3)THEN 1
440
1430 IF JAC(HVV(1))THEN JA=-
1 :: JC=15 :: GOSUB 370
1440 FOR WD=1 TO 3 :: IF HVV
... -13 ' % WI=WI+1
.:.: NEXT WD :: IF JA THEN W
I=10*WI
1460 NS=-(HVV(1)=HVV(2))-(HV
V(2)=HVV(3)):: IF NS>0 THEN
WI=WI+R(NS,HVV(2))
1470 IF DO THEN WI=2*WI
1480 RETURN
.:%: REM ### GAMBLE ROUTINE
. .. DT=1 :: 6V(1)=2*WI :: 6
% 0 :: 6V(3)=INT(3*WI/2):
: 6V(4)=INT(WI/2)
1510 KEY=0 :: NG=NG+1 :: GOS
UB 350 :: DISPLAY AT(24,1):"
.. .";
:.:: FOR LI=1 TO 3 :: CALL H
%::: LI,16,96,17):: NEXT LI
.::: RR=RR+1 :: IF RR)4 THEN
RR=1
:%: DISPLAY AT(6PY(RR),6PX(
 ..:STR$(6V .. -:
.. CALL 6L.%: -4000,150+50
1-.:)
1.. IF KEY()ASC("S")THEN CA
LL KEY(0,KEY,CH):: GOTO ....
1570 IF DT=1+RND(2)1DT :: FOR
DE=1 TO DT :: NEXT DE :: IF
DT)150 THEN .. :
1580 CALL HCHF...- "(RR),6PX(
RR)+2,96,4):: .. 1530
.. :: FOR LI=1 iu 3 :: CALL H
.. LI,16,32,17):: NEXT LI
1600 6W=6W+6% -  WI :: WI=6
V(RR):: CALL  %J(-1,150+50
*RR,2):: RETURN
1610 REM ### END OF GAME
1620 .%..: CLEAR :: CALL CHAR
SET :: .-:. SCREEN(2)
1630   .. AY AT(5,1):"AMOUNT
OF - % :"
1640 -.  .. * $ ####.##"
1650  .. AY AT(7,3):"PUT IN
:"::: uisPLAY AT(7,18):USING
1640:XI
1660 DISPLAY AT(8,3):"GOT BA
CK:"::: DISPLAY AT(8,18):USI
NG 1640:MO/4
1670 DISPLAY AT(9,3):"MAX AT
ONE TIME:";::: DISPLAY AT(9,
18):  % 1640:MM-4
1680   % AT(1.%-:.-:"WON B
Y 6AMo.'% :"::: .. AY AT(I
0,18):!  % 1640:6W/4
1690 :.  AY AT(12,1):"NUMBE
R OF - .  ";:NH
1700 ::. AY AT(13,1):"NUMBE
R DF .  S:";NG
1710 :  %Y AT(14,1):"NUMBE
R OF - .. :  ";NT
1720 -. : .%
1730 - % -:.N PROGRAM
1740   41D 'OFF CHAR
1750   650 ! :  N
1760 .  840 !: :  T .
1770 -% -2*HB-(MO/11)+1 GOSUB
950...:..950,1080
:. .:. .. AY AT(5,9):: CA=-(
.. :.ASC('I"))-2*(GET=ASC("P"
))-3*(GET=ASC("E"))
1790 ON CA+1 GOTO 1770,1800,
1820,2020
1800 NI=NI+1 :: IN=4 :: GOSU
B 250
1810 GOTO 1770
1820 IN=-2 :: GOSUB 250
1830 GOSUB 1140 !SPIN WHEELS
1840 IF DO  % 1840
.: .. .. .:: .. :
.:.: .. :   .. :.
.:.: ir nr$.;/-n1$ ok nr$(2)
-%# OR HF$(3)=H1$ THEN 1890
.%.% :: WD=1 TO 3 :: IF WI)
% incN 1910 ELSE 1760
1890 HB=-1 :: IF WI)LW THEN
1910
1900 DISPLAY AT(5,9):" YOU L
OST" :: .."%: SOUND(-600,200,
2)::  ' :760
1910 ..  :Y AT(5,9):"YOU WO
N";::: uisPLAY AT(5,17):WI;::
FOR DU=1 TO WI :: CALL SOUN
D(-50,500,1):: CALL SOUND(1,
500,30):: NEXT DU
.. :: IF -.: THEN HB=-1
1930 GOSUB 1120 !GET INSTRUC
TION
1940 CA=-(GET=ASC("H")AND NO
T HB)-2*(GET=ASC("G"))-3*(GE
T=ASC("C"))
:. : ON CA+1 GOTO 1930,2010,
: .1960
. : IN=NI :: ' : 250
. : IF WI)MM : % MM=WI
1980 GOTO 1760
1990 HB=-1 :: GOSUB 1500 !6A
MBLE
2000 IF WI)0 THEN 1910 ELSE
1900
2010 WI=0 :: GOTO 1760
2020 GOSUB 1620 !END
2030 END
```

Good Old Days-Part II

(Part 2 of a 3 part series on the early days of the TI-99/4A. Part 3 will appear next month.)

After I had amassed my 100-plus programs on tape (mostly typed in from "99er magazine and some early books), I like so many others wanted to expand my computer masxstery. My TI appetite was voracious.

I bought every TI book available, which in 1983 was a considerable number. These included such masterpieces as "The TI-99/4A in Bits & Bytes", "Elementary TI-99/4A", "Terrific Games for the TI-99/4A", "TI-99/4A Favorite Games Explained", "Programs for The TI-99/4A", and the series of books published by Sams and Compute Magazine. Regina's (Cheryl Whitelaw) book, "Programmers Reference Guide to the TI-99/4A" was always at my side. She wrote such programs as Homework Helper, Name that Bone, Typing for Accuracy, Civil Engineering Fundamentals, Harried Housewife, Dodge 'Em, Maze Race, San Francisco Tourist and many others. There were others of course, but these were some of the most popular. What astounded most of us Regena groupies was the depth of her knowledge. She seemed to know an incredible amount about computers (and a lot of different kind of computers) but she also knew so much about music, children, electrical engineering, geography, and just about any topic a staff of college professors would know.

That was in 1983. Regena was already in her third year of TI publishing when she moved to Compute Magazine. So I began a subscription to continue getting her monthly column. I got more out of that magazine, too. Tutorials of all kinds. Programs galore. It was TI's Cloud Nine! In that same year, Regena began writing for Enthusiast as well.

Boy what a year! Mark Leyton began his wonderful Unofficial 99/4A magazine. K-Power and Family Computing also made their debut and supported the TI-99/4A. I subscribed to them all - 6 TI magazines a month (7 if you include the non-monthly Smart Progammer). Super 99 Monthly and Minimag 99 would not come out for another year.

I find it hard to believe that in 1983 the final 4A classic book - "The Last Whole TI-/994A Book", by Paul Garrison arrived. It was published by Wiley Press in 1984 and still stands as one of the best books ever written for the TI-99/4A. (Do not confuse this book with "The Last Word on the TI-99/4A", which is probably the worst book in my TI library.

It's even harder to believe that the magazines mentioned above no longer exist (or at least do not carry any TI stuff.)

As magazines like Compute and Family Computing dropped TI (which still had the largest home computer ownership of all) more than Apple, Commodore, and IBM Jr. combined, many of us wrote in protest but to no avail. After all, we were still buying disks, drives, monitors, printers and lots of advertised

things. The magazines that dropped TI even carried information for the Adam, the Peanut and Timex (do you remember them?)

Micropendium was first published in February 1984 (originally entitled Home Computer Compendium) and to this day is the only regularly published magazine devoted entirely to the TI-99/4A (recently Geneve was added). Editor/Publisher John Koloen said this of his new magazine, "It is a conduit, a source of information and a vehicle for the dissemination of information." It has certainly lived up to that criteria. I would find it difficult to be a 4A owner without MICROpendium and user groups. The only other magazine still supporting our community with a very informative monthly column is Computer Shopper.

It's sad to look through that first Compendium. All the advertisers except one are gone: Software (Megaworld), TJ Software (Kandy Kong), THinc (Colors), C.A. Root (On Gaming), Maple Leaf (Sky Diver), Silicon Valley Software (S.A.T. Verbal Sectionl, Machine Shop Software (Cassette Indexerl, Larry Vision (Quackers), Program Software (Personal Enrichment), Microworld (Snac Man), Soft Relations (Super Speller), DCH Software (Home Budget), Callcar (Emotional Helath), Software Programs (Starchip Concord), TI Books and Software.

The one advertiser in that issue who is still with us is TIGERCUB SOFTWARE, owned and operated by Jim Peterson (Mr. T.I.).

And that brings us back to 1983 (before the orphaning) when we were still high on TI and user-group business was booming. For all the other things that were going on for us (and they were numerous) the big department stores and book stores and software stores all carried 4A items on their shelves.

User-group newsletter articles were not as sophisticated as they are today. The big exception was the "Tip from the Tigercub" monthly column from Jim Peterson.

Nobody did what he did. His articles are still being published today in many newsletters and have opened the door to understanding. One was either a techie or a dummy. But Jim made us all feel intelligent. He pulled us up. His explanations and his "experiments" and his enthuiasm came out in every article. He always stayed far enough ahead of us to challenge us completely, yet he never talked down to us.

From his very first article: "Are you tired of that blankety blinking black cursor?". Well this won't work in BASIC but try it in XBASIC

1 CALL COLDR(0,11,1)

I tried it and it worked and then I fiddled with the numbers and I finally understood what happened.

This is the way Jim taught, for teaching is what he certainly did (and does). His classroom included thousands and thousands of pupils.

He taught us to use Line 0 and FCTN/X (or E) instead of EDIT and how to

use REDO for expanding program lines and how to slash the zero and how to highlight operators. He taught many more things: You can type RUN"DSK1.FILE" without any spaces and it'll work; that typing the double colons in XB lines without spaces before or after won't matter. Do these sound mundane? Not if you didn't know any of that stuff and none of us did.

His teachings were so natural, so filled with personal experience, that you couldn't wait to try the thing Jim had just found out. Here's another from the first "Tips": "Have you ever been typing in a program and the computer suddenly jumped back to the title screen, and you were sure that you didn't have a finger anywhere near that infernal QUIT key? So if you don't have anything valuble in the computer right now, try pressing FCTN, Space Bar, H and N all at the same time." Oops! Another useless bit of info - try FCTN, 5, 6, and 7 all together. Break!

My vision of him was a young kid (maybe as old as a college student) doing improvisational computing, the way Art Tatum played the piano; somebody with great knowledge and understanding exploring human/machine potential.

I didn't find out until later that Jim is a grand father and that he bought the 4A because he liked the keyboard!

It didn't matter. Genius is genius no matter what the age.

He used to close off those early columns with Happy Hackin' until hackers got a bad reputation by a few pirates and vandals, then he closed it with Memory Almost Full. Too bad, in a way. I think of Jim as the Ultimate Hacker, in the real sense of that word. I don't know anyone who knows more about BASIC and XBASIC than Jim.

Way back in 1983 he did a lot more than give us little "useless tips". Most of the tips are the most "useful" things I ever learned for my computer. There is no Jim Peterson for Apple, Commodore, Tandy or IBM.

Each article contained at least one original type-in program. While other programmers and writers were making bundles selling their stuff to commercial magazines and software houses, Jim GAVE his monthly column away in exchange for the user-group newsletters. He probably has the largest library of TI written material in existance. Whenever any item in his column, large or small, came from another source, Jim always credited the originator. His generosity is known throughout the entire TI World.

His programs for which he charges $3.00 apiece, include a discount for future purchases. Other programmers at that time were charging $20.00 and $30.00 for programs that were not nearly as professional. Now his programs are just $2.00 or even less in multiple orders.

The first batch of four programs I ordered came back with seven programs in the package. Jim always puts in a little "extra" in each order. He still

TI DOS
By Rick Felzien
West Jax 99ers

As I was doing my research in preparation to write a series on the use of the Advanced Diagnostic Program from Miller's Graphics a thought occurred to me. I realized that perhaps not everyone has had the opportunity to become as familiar with the TI-DOS as I have. To use the Advanced Diagnostics effectively, one must understand a good bit about what is on the disk and what that data means to the system.

We shall begin by examining sector zero. This is where the computer tells whether the disk is initialized and keeps track of such pertinent information as sectors used/available, how many sectors are formatted, whether the disk is single or double sided, and whether it is formatted as single or double density. Let us take a look at sector zero.

What you should see is shown in Figure 1 using the Edit Sector Command. The information is used in either byte size or word (2 byte) sized units to keep track of the data.

```
5445535420202020202005A01244
534B202802020200000000000000
00000000000000000000000000000
00000000000000000000000000000
03000000000000000000000000000
00000000000000000000000000000
00000000000000000000000000000
00000000000000000000000000000
00000000000000000000000000000
00000000000000000000000000000
00000000000000000000000000000
00000000000000000000000000000
00000000000000000000000000000
00000000000000000000000000000
00000000000000000000000000000
0000000000000000000000000FFFF
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
FFFFFFFF
```

Bytes >0->9 are for the disk name. If it is less than 10 characters long the trailing characters are blanks. (To keep things consistant hexedecimal numbers which is how all data is used and displayed on the screen will appear with a preceeding greater than character [>] in this article.) Bytes >0A->0B denote the number of formatted sectors, >0168=SS/DD, >02D0=DS/SD, and >05A0=DS/DD.

Byte >0C is the number of sectors per track, >09=9, and >12=18(for DD).

Bytes >0D-0F are the letters DSK which the controller looks for to see that the disk is initialized by a TI compatible controller.

Byte>10 is used for the protected/not-protected code which was used by TI to protect some of their disks in the beginning. >20 (space char.)=Unprotected, 50 (letter P)=Protected.

Byte >11 denotes tracks per side >23=35, >28=40.

Byte >12 designates the number of sides formatted either >01 or >02.

Byte >13 tells us the formatted density either >01=SD or >02=DD.

Bytes >14->55 reserved-not used-all zeros.

Bytes >38->FF are the allocation bit map. This is what tells the disk controller which sectors are used and which are free. Each byte controls 8 sectors and this is controlled by the fact that each bit of the particular byte is either a 1 for used or a zero for unused. On this particular disk there 3 sectors used.

In the allocation bit map the format is as follows:
SS/SD used bytes >38->64, SS/DD or DS/SD uses bytes >65->91, and DS/DD uses bytes >92-EB, and >EC->FF are unused and are formatted to all F's, or used with all bits on. On initialization the actual bit map bytes are set with all bits off or zeros.

For instance byte >38 controls sectors 1-8 as follows:
```
bit nos-  7  6  5  4  3  2 1 0
sector nos- 7  6  5  4  3  2 1 0
```
Byte >39 would be:
```
bit nos-  7  6  5  4  3  2 1 0
sector nos-15 14 13 12 11 10 9 8
```
And so on.

To do the mathematics for finding which sector is controlled by which bit of which byte is easiest to convert to decimal, do the math and then convert back to hex.
```
Byte no - 56=start of 8 sector group
   >38 - >38=0=sectors 0-7
   56 - 56+0=sectors 0-7
```

```
sector no/8+56=byte no    remainder*8=bit no
   137/8+56=73.125           .125*8=bit no 1
   34/8+56=60.25             .25 *8=bit no 2
              and so on.
```

(Eds note:-refer to our program in the March '88 and April '88 issues in which we listed a conversion program.)

When I started using Diags. and had gotten fairly familiar with the DOS, I often wondered how the Check Disk (CD) command kept track of sectors that are mapped bad. This is easy when the disk is freshly initialized and contains no files as they are designated as being used in the bit map. But I said to myself how is this done after several files are put on the disk and a good bit of the bit map is used. How did it keep the bad sectors separated from those that were used. I looked at both a good disk and one with several bad sectors and could see no special coding, even on the track header data. Being curiuious I called M.G. and the people there were very courteous and helpful. What happens is that after there are files on the disk, and if there are some bad sectors, the Diags. programming does a compare type operation. It checks the allocation bit map and then checks the file directories for the sectors that the files occupy and any sectors that are designated as used and are not included in the file directory information are considered to be mapped bad.

In case any of you wondered how I got the Diags display printed, here is how to do it. With the OD command (Output Device) you can select DSKn.filename and Diags. will dump the screen to a Display/Variable 80 disk file that can be merged into the TI-Writer file.

Sector 1 is the directory link and tells the disk drive where to look for the directory sectors for the files.

Gram Kracker

Ted Peterson is forming a group of interested members in exploring the use of the Gram Kracker. Although this piece of hardware is no longer manufactured, many of our members have one. If you are one of those you might want to take advantage of joining this group. See Ted at the meeting to learn more about these planned meetings.

TI DOS - continued

```
                      0013001A0003001400004000050006
                      001E0018001C001B000700190008
                   x  001500090000A000B001D000C0016
                   e  0010000D0011000200000E0012000F
                   H  000000000000000000000000000000
                   :  000000000000000000000000000000
      Track        y  000000000000000000000000000000
      Sector       a  000000000000000000000000000000
      Display      l  000000000000000000000000000000
                   p  000000000000000000000000000000
                   s  000000000000000000000000000000
                   i  000000000000000000000000000000
                   D  000000000000000000000000000000
                      000000000000000000000000000000
                      000000000000000000000000000000
      Drive           000000000000000000000000000000
      Side            000000000000000000000000000000
      Byte            000000000000000000000000000000
                      00000000
```

Although the files are placed on the disk in the order that they are saved, the link numbers are shuffled to give correct positions for the alphabetical order which shows up on a catalog operation.

This data is stored in one word or two byte blocks. The first alphabetical or A program on this particular disk has it's directory link on sector 0013 or >13, but 0013 is the first number thus denoting that it is the first in alphabetical order.

When a file is deleted, it is not actually erased. The link number is removed from this sector and the bit map on sector zero is changed, but the data is still on the original sectors and is merely overwritten as more files are added to the disk.

Sectors >2->22 are called the File headers and sometimes called the File Descriptor Blocks.

```
                      5055555A4C455220202000000100
                      000D8E0000000000000000000000
                   x  22C0000000000000000000000000
                   e  000000000000000000000000000000
                   H  000000000000000000000000000000
                   :  000000000000000000000000000000
      Track        y  000000000000000000000000000000
      Sector       a  000000000000000000000000000000
      Display      l  000000000000000000000000000000
                   p  000000000000000000000000000000
                   s  000000000000000000000000000000
                   i  000000000000000000000000000000
                   D  000000000000000000000000000000
                      000000000000000000000000000000
                      000000000000000000000000000000
      Drive           000000000000000000000000000000
      Side            000000000000000000000000000000
      Byte            000000000000000000000000000000
                      00000000
```

Bytes >0->9 make up the filename using 1 to 10 characters.

Bytes >A->B are zeros and are not currently used for data.

Byte >C tells the controller the file-type. If the file is protected, the value of 8 is added to the unprotected code number.

Type	Unprotected	Protected
DIS/FIX	00	08
DIS/VAR	80	88
INT/FIX	02	0A
INT/VAR	82	8A
PROGRAM	01	09

Byte >D denotes the number of records per sector. This number equals the sector size (256 bytes) divided by the record length

— (>100/>50=>3 or 256/80=3). Program files always=0. DIS/FIX or INT/FIX 40 =>06, DIS or FIX 60 = >04, DIS or FIX 80 = >04, DIS or FIX 80 = >03 etc.

Bytes >0E->0F equal number of sectors in the file (not including the file descriptor). This is the cataloged length minus 1. Byte >10 is called the end of file offset. For variable length files and programs this byte lets us know the number of bytes in the last sector of the file that are used. It also indicates which byte number is the EOF marker. For fixed length files this is always >00. The last byte of the last file sector is an end sentinel - AA for programs and FF for all other types of files.

Byte >11 gives the logical record length. FIX or VAR 40=>28, 80=>50, 163=>A3, and 254=>FE. Bytes >12->13 are the number of fixed length files or else the number of sectors in variable length files and are not used by programs. The bytes of this two byte block are reversed so that >0500 is actually >0005.

```
                      44454D4F2D312020202000000B803
                      0006C25006000000000000000000
                   x  2C30009150000000000000000000
                   e  000000000000000000000000000000
                   H  000000000000000000000000000000
                   :  000000000000000000000000000000
      Track        y  000000000000000000000000000000
      Sector       a  000000000000000000000000000000
      Display      l  000000000000000000000000000000
                   p  000000000000000000000000000000
                   s  000000000000000000000000000000
                   i  000000000000000000000000000000
                   D  000000000000000000000000000000
                      000000000000000000000000000000
                      000000000000000000000000000000
      Drive           000000000000000000000000000000
      Side            000000000000000000000000000000
      Byte            000000000000000000000000000000
                      00000000
```

Bytes >14->1B are all zeros and are not used (reserved for future use).

Bytes >1C------These keep track of the blocks of sectors that the file actually occupies on the disk. This is done in 3 byte blocks and are not read as they appear in the block. Nybbles 4,1, &2 are the beginning sector and Nybbles 5,6, &3 are the number of sectors occupied by that block of the file.

The following is an example of how to read these bytes in the case of a badly fractured file which is in five segments on the disk. This does not usually happen, but will show you how to read this block of data effectively.

Sector address (hex)	address contents (nybble) 12 34 56	start sector (nyb) 412	addt'l prior sectors (nyb) 563	logical end sector	size of block of sectors	Subtot.
1C 1D 1E	23 30 00	023	003	026	4	4
1F 20 21	31 40 00	031	004	031	1	5
22 23 24	58 50 00	058	005	058	1	6
25 26 27	5A 10 01	05A	011	065	12	18
28 29 2A	67 60 01	067	016	068	5	23
2B 2C 2D	84 80 01	084	018	085	2	25

Total data sectors=25
Directory sector + 1
Cataloged sectors =26

"23 Skidoo!"

I guess the use of slang today is accepted and used more than ever. In doing a little research for this month's english essay, it seems that good old Ralph Waldo Emerson found the word 'jawing' acceptable while Ambrose Bierce denounced slang as 'the speech of those that rob the literary garbage carts on the way to the dump.'

Strong words indeed and probably Emerson is more with it than Bierce by todays standards, yet the use of slang carries with it a responsiblilty. "Drug-store cowboy," "cat's pajamas," "heebie-jeebies", "make whoopee" or "23 skidoo" will immediately date you.

If you want to communicate with people today, you either ought to be current with your use of slang or be content with less colorful language. Whether writing a letter or speaking, if you use slang expressions, don't get too far ahead of your audience or the slang words may do you in.

To be avoided, no matter what the temptation, is to try and bridge the generation gap. As one advertising executive warned:

"Those of us who are required to establish some form of communication with teen-agers should stick to our particular idiom of American English or risk making damn fools of ourselves."

The surest way for an advertiser to make sure he's turned off or tuned out by teen-agers or other distinct groups is for him to portray an obsolete fad, dance or mode of dress, or to misuse the specialized idiom of the day."

While slang properly used adds color and strength to what might otherwise be a pedestrian style, care must be exercised so that no matter how far out the expression seems, it is at least understood by those to whom intended.

Every business and profession acquires it own jargon and within a business it acts as a kind of shorthand, presumably saving time and thought. Beware though the man who tries that jargon outside of his own business.

Charles Ball, editor

Newspaper Drive

Don't throw away any of those old newspapers. We're going to have a paper drive very soon and your donation of newspapers is needed.

There will be more information about this important activity in next month's "WordPlay", so in the meantime pile them up instead of throwing them away.

GENIAL TRAVelOR

The second and third issues of Volumn II, The GENIAL TRAVelOR, has arrived and will be for sale at the November meeting. As usual they are $6.00 each and each issue includes two SSSD disks packed full. You can purchase them from the Editor prior to the beginning of the meeting or during the break.

Program

At our meeting on Tuesday November 1st., our own Jim Smith will be on the program.

He is going to explain how to use the Hard Disk that has been developed for our computers. This program is one of the hilights of our club year and you will want to be there.

He will not only explain the features of this harware, but will also demonstrate how it is used.

PUNN is fortunate to have among its members, talent of this kind and we appreciate the time and effort that goes into producing a program of this type.

Ted Peterson needs to hear from you in order to plan future programs. What are your interests? Games, Utilities, Data Bases? Let Ted know by calling him at 244 1587 and who knows but that your favorite program could be the topic for the next meeting.

Good Old Days-continued

does. He has over 3000 Public Domain programs which he shares by putting them as bonuses on any disk order. These programs are well written and worth owning.

I was surprised by the bonus, as no clue was given by him that I was going to get more than my money's worth. There was not a single program that I ever got from Tigercub that did not exceed my expectations. Many I ordered for my 5th. grade class and are still popular (like BAZOO and MECHANICAL APPTITUDE TEST). It would be hard for me to pick a favorite because there are so many varieties of programs: educational,

music, utilities, games and so on.

About three hours ago I saved this textfile and loaded up some of Jim's disks. I played the fiendish SQUINCH and the diabolical SCRUM and went through his HANDY DANDYS and SPEEDER READER and - I noticed it somehow got to be 2:10am. I have to teach tomorrow and Jim Peterson, the friend, did it to me again!

Next day!

When I reread the above, I got to thinking about all the other TIGERCUB programs I didn't mention. The most important, for me, are the three "Nuts & Bolts" disks containing over 300 files.

When these started coming out a few years ago, it was a "TIPS" maniac's delight. Here was a discovery collection that every TI learner dreams of. These files can be merged into any XB program to produce remarkable results. Because of incremental line numeration, multiple subs can be called into single programs. They are efficient, neat, incredibly easy, and remarkably creative. And lots more. Send Jim $1.00 for his catalog, refundable with first order to:

TIGERCUB Software
156 Collingwood Ave.
Columbus, OH 43213

PUNN Disk of the Month

EPSONSET X-BASIC
Program to help you set up your printer. Will work with Star NX-1000 and other Epson compatible printers.

MASH-XB X-BASIC
Music-main title theme from the novel MEATSH. This program has animation and the words to the theme for sing-along.

NINE-CUBES X-BASIC
NINE-CUDOC TI-WRITER
Rubix Cube type game. Line up all the colors in a certain order on all sides of each cube by using the coordinates shown on the screen. The documents and instructions are included.

TINY/CAL X-BASIC
Prints a tiny calendar for any year from about 1700 to 2099. Uses the subscript mode and works fine with Star 10X and NX-1000. Should work well with other printers. This program can be modified if you want to.

TYPE1, 3, 4 X-BASIC Typing tutor by C Regena. Teaches you how to type by using graphics and sounds with demo lines for you to type in. I hope it is all there. No TYPE2 was with this when we received this disk from C.R.

WOODSTOCK X-BASIC
Christmas program-Woodstock, Snoopy's littel birdfriend, is left alone on Christmas eve. Snoopy left him a gift that he has to retrieve from Snoopy's doghouse. Good animation with translations of Woodstocks chirping printed on bottom of screen.

ALL GENERAL MEETINGS ARE HELD
ON THE FIRST TUESDAY OF EACH
MONTH, AT THE PGE BUILDING
3700 SE 17TH AVENUE
PORTLAND, OREGON

!! NEXT MEETING DATE !!
NOVEMBER 1ST. 1988

The PUNN Newsletter

WORDPLAY
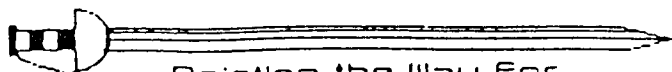
Pointing the Way For   Ri
Users of TI's 994/A Computer

November 1988-Volumn 7-No.11