SouthWest Ninety-Niners

January 1996

P.O. Box 17831 Tucson, AZ 85731
(520)747-5846

President's Notes

by BJ Mathis

We had a very enjoyable Christmas Party on December 5th. 24 SW99ers attended including many family members. The gift exchange was even more fun than last year! One sobering note was the absence of John Hale. Alzheimers has claimed him. Maxine was no longer able to care for him, so he is in a home for alzheimer patients. Jack and I delivered the gift from the SW99ers to Maxine. We enjoyed our visit. enjoyed our visit, it is obvious she misses John.

Les informed us the band that was practicing at the VFW during our November meeting will be practicing every Tuesday night. The VFW offered to allow us to have our meetings on Thursdays. A survey of members of members at the Christmas party allowed us to ask Les to make arrangements to have our meetings on the first Thursday of each month. The January meeting will be January 4, 1996 at 7pm.

Election of Officers

January 4th we will elect the officers for SW99ers 1996. All members need to consider their choices for the coming year. Nominating committee members were chosen at the November meeting. During the month of December the Nominating Committee contacted local SW99ers to find out if they will run and who they would like to have run.

I don't plan to run for any office this time, as I have been President since 1987, eight years is enough at a time! So please consider yourself when thinking of officers for 1996. At this time I plan to continue as newsletter editor at least for a while, but as I sit here trying to get the newsletter done during my "Christmas vacation", I don't feel much like I am on vacation!

FestWest '96

The most important dates for SW99ers to remember during the coming months are February 16-18th - FestWest '96 in Tucson! We are preparing a list of Raffle Prizes that includes a Myarc HFDC and 40meg hard drive, a 24-pin printer, a Horizon RAM disk, a SuperAMS (of course!), and possibly even more prizes. We have already begun selling Raffle tickets. You can get your raffle tickets by writing to SW99ers, P.O. Box 17831, Tucson, AZ 85731. As in the past, purchase 10 Raffle tickets, your admission to FestWest '96 is free!

We are planning to have the set up for FestWest '96 on Friday evening. We expect to have the hospitality room open that evening as well. The actual Fest will take place on Saturday. Vendors will be encouraged to wait until Sunday morning to pack up. This will allow them to join in on the usual gab-"Fest" that usually occurs Saturday evening. Sunday will be available for those who need to get home, or for sight-seeing for those who get a holiday on President's Day!

Fest West '96 Vendors

Don O'Neil, Western Horizon Technology, has expressed his intention to attend and has added to the list of Raffle prizes. Competition Computer will be represented by their new owner. The new owners of TexComp, also plan to be at Fest West '96.

We are looking forward to another great FestWest in Tucson. It is a chance to catch up with new things in the TI community and talk to friends we have made over the many years.

Tom's Observations

by Tom Wills

Another Year!
The subject for the next meeting in January is the election of officers for the 1996 calendar year. So far we have only one dedicated user group member who has come forward to offer his services.

What about the rest of the user group? Does it always have to be the same people every year who run the SouthWest Ninety Niners User Group? Is there no one else in this organization who is qualified to be a leader? I doubt it. Many of you can do as good a job of running the SouthWest Ninety Niners as the present slate of officers.

DJ, Mike, and I are getting overburdened with our everyday activities. Plus we are starting to feel burnout. None of us will drop out of the group, we'll still be there to lend a hand. But how about some new leadership?

Call Mike Doane, Tom Buick, or me if you are willing to lead the SouthWest Ninety Niners in 1996.

Internet Surfing
I received a message from Josh Spatz. In the message was a Hyper Text Markup Language (HTLM) file to access TI-99/4A related World Wide Web (WWW) sites. This TI Home page is part of Rick Polvika's home page. If you wish to visit Rich's home page, that address is w3.gwis.com/~polvika/index.html. There is a link to the TI-99/4A home page from here.

I'd like to publicly thank Josh Spatz for sending me this information. I am passing this along as this is information I want the whole TI community to know about!

The rest of this section is going to drive our editor nuts due to the use of special characters, such as the Tilde.

Some things to note are to access these sites requires a PC (I know, Boo, Hiss!) using a web browser such a Netscape. All the addresses are preceded by "HTTP//:".

The first site address where the other sites can be accessed is The TI99/4A Home Computer Page (w3.qwis.com/~polivka/994apg.htm). My experience with this page, however, is that it loads very slow. I am using a 486-DX4 computer running Windows 95. Many factors beyond my control may be causing this slowness. You may have better response time than I'm getting.

The first thing on this page is a link to the TI Newsgroup (COMP.STS.TI), and Jim Reiss' TI-99/4A FAQ (Frequently Asked Questions).

Next is some information of the TI-Chips User Group of Fairview Park, Ohio. Also included are some screen captures of various TI programs, such as Parsec, Munchman, Return to Pirate's Isle, Tombstone City, and Tunnels of Doom. As these are very graphical, the loading time will be slow.

Also listed is a TI-99/4A File Transfer Protocol (FTP) site, which is at solutions.solon.com (in the /pub/ti99 directory). Also included are my internet email addresses:
twills@indirect.com and twills@pima.gov
for those TIers who are interested in getting themselves included in the TIers Internet Address List.

Links are also included to a FTP site to obtain two TI-99/4A emulators (COMP.EMULATORS.MISC FAQ). Then there is the link to Bruce Tomlin's TI-99/4A's module listing, which can be found at:
FTP//:ftp.crl.com/users/bt/btomlin/carts/ti99
Please note this site does not start with "http//:", but with "ftp//:".

A set of links to TI vendors is coming soon, or so it says. This will be nice.

Some related TI links include The TI-99/4a Page at sys00.ti6.tu-harburg.de/~ti6hk/hobby/ti/index/html This site, which is in Germany, contains the Internet TI Mailing List and the Cartridge list of Bruce Tomlin. Plus things such as a Prime generator algorithm and Tricky stuff to speed things up. Check this site out!

Another interesting site is the Obsolete Computer Museum at www.ncsc.dri.us/fun/user/tcc/cmuseum.htm. Besides information on the TI, this site contains information about computers such as the Osborne 1, the Altair 8800, and the Magnovox Magnum 16-cd, just to mention a few.

One site which you would expect to be interesting is the Texas Instruments home page at www.ti.com. However, they don't even acknowledge the 4A. This is not a site to waste your time on, unless you do like I did, and that was complain to TI via their feedback page about this lack of 4A presence.

Another site is the Video game Advantage at www.io.com/~vga2000. You will find some TI-99/4A references here. Plus they have a For Sale or Trade and a Want list. The lists on this site will change from time to time.

There is also a link to a well known TIer, who contributed much to the TI community, Chris Bobbitt. His home page is at:
www.cais.com/fmg/bobbitt/clb.htm
Drop in to Chris' home page and say "Hello."

One thing I really like is this presence on the internet really shows that the TI-99/4A community is still a strong an viable entity.

Epilogue
I hope that you all had a great Christmas, and a good New Year celebration. This marks the begining of another year as orphans. Who ever thought ten years ago we'd still be here. My New Year's resolution is to keep the TI-99/4A and Geneve 9640 computers in use for another year.

This includes supporting Fest West '96. I'll be looking forward to seeing all of you at the Best fest in the West. Fest West '96!!!!

January - Disk of the Month

Compiled by W. Leonard Taffs

WINDOW:1-6 by Leonard Taffs, along with its docs, takes up 239 sectors of this month's disk. It is a file-reading program capable of displaying 6 files at a time, displaying them record by record, in two 3-window-per-screen displays. It un-Arcs to some 487 sectors, so those with 360 disk drives will need two disks. This is the same version that was published in a recent MICROpendium. Be sure to read the docs which explain when a need for doing a CALL FILES is necessary and for explanation of some help that is not shown on-screen when running the program. Anyone is welcome to copy or modify this program but please give source credit to SouthWest 99ers. Thank you.

As this month marks the anniversary of the passing of one of the most prolific of T. I. "gurus", Mr. Jim Peterson, the remaining free sectors on this disk is used for 2 programs of his and one of his TIPS columns, as a tribute to his memory. The programs are TAKEAWAY, an elementary math program, intended for the continuation of his Math series, which he abandoned for lack of interest shown in his doing so, and his short WORDCOUNT, which is listed as WORDCOUNTP on this disk.

For WORDCOUNT program comparison purposes, two modified variations of the DAVIES WORDCOUNT program, are included along with *READWORDC, which gives some background on these WORDCOUNT programs.

We are indebted to Dave Swartz of the ROCKY MOUNTAIN 99ers for making TIPS #39 and TAKEAWAY available. The TI sound synthesizer is optional in TAKEAWAY. For a comment on TAKEAWAY, see Feedforth column.

TIPS #39 is included on this disk in case some new members may not have seen any of this series, or had the opportunity to study some of his programming techniques. The last TIPS seen by this writer was numbered #70, which conveys to us what an enormous output Mr. Peterson produced. Write to the LIMA Ohio U. G., now in possession of his library, which formed his TIGERCUB Software collection, for information on available files and programs.

Forget About Assembly;

C is What You Need

by David Ormand - SW99ers

As processors become more and more sophisticated, particularly with the RISC machines with features (like look-ahead branching) which are not exactly easy for humans to understand or use, it becomes less and less feasible to program computers in assembly language. It hurts me to say that, since I am by preference an assembly language programmer, but those are the facts.

Assembly language seems to be the domain of special stuff, like boot or startup code, or time-critical stuff like interrupt service routines, or writing operating systems and compilers, but not for general-purpose application writing. The inescapable trend is to high-level languages, and in the market today, there are only two real choices: Pascal-like languages, like Pascal, Modula-2, and Ada, and C (and its extension C++). The Pascal-like languages, in practical usage, are really represented by Ada, which was developed for the defense industry but is not in heavy use anywhere else except for Europe. The predominant language in science and industry, then, is C, and therefore it is almost mandatory for anyone who intends to program computers for anything more than pleasure to learn the basics of C.

The good thing here is that, while TI doesn't have Ada (it is bulky, to say the least), it does have Pascal and C. The last time I looked (without keen interest, I must confess), Pascal required the use of the P-Code card, which is not a universal TI peripheral. C, on the other hand, results in E/A Option 5 program files, which almost anything can load. So you can use C on your TI, not only to learn the most commonly used programming language in the world, but also to create fast, compact, and easy-to-use TI programs.

Let's get started. What you need are the TI disks for C99 version 4.0 by Clint Pulley. Versions for the TI-99/4A and MDOS on the Geneve are available. What you find on the disks are:

- Documentation
- Compiler program (C99C, D, E)
- Include files (more about those in a moment)
- Library files

Other things you need are an editor (like a TI-Writer clone) and an Assembly (like the ancient TI assembler, but the RAG assembler is better).

Okay, run your editor program. C programs are just text files, like letters and stories, so you can use any old editor that writes a DV/80 text file to write C programs. Enter the following program:

```
#include "DSK1.STDIO"
extern printf();

main()
{ int i;
     for (i=0;i<5; i++)
     printf("Hello number %d\n",i);
               /* Print the numbers out */
}
```

Then save it as file DSK2.TEST_C". Now let's look at this simple program.

The first line is an "include" statement DSK1.STDIO is an "include" file. If you want you can load DSK1.STDIO into the editor and look at it; it is a series of C-language lines, When the complier program sees an "include" statement, it loads the file and compiles it just as if it were part of the program. This makes it rather like as "COPY" command in assembly language.

C is a library-oriented language. Unlike BASIC, where all the tools you need are always available - like SIN() or SEGS() - in C you have to load the library with the functions you want. Plus, you have to tell the compiler that your are using library functions and features. DSK1.STDIO has a bunch of constants defined to allow you to use standard input and output, such as files, keyboard, or display. The next line, "extern printf();" tells the compiler to use the function printf() out of a library file. Note that on other machines the printf() function will be part of the STDIO include file, so you wouldn't have to say "extern". This is just a TI-C99 thing. Also, on other machines, the STDIO include file is called "stdio_h". Which is why you might want to look on your disks and see if you have STDIO or STDIO_H, and use the correct filename in the program above.

The "extern" is a flag to the compiler that the symbol "printf" is a function defined in an external file. This is exactly the same as "REF" in assembly language.

At this point in your experience with C, it is enough to know that everything is functions. That's not exactly true, but right now it's good enough. The "printf()" is a function that prints formatted text strings. And "main()" is the function that is run when the program is started, just as if it were the lowest line number in BASIC. Notice all the functions have parentheses after them, just like the BASIC functions. The arguments would go inside the parentheses, just like BASIC, and if there aren't any arguments, you have empty parentheses, just like the "main()" function here. Note that when you use "extern" in a C99 program, you use empty parentheses to tell the compiler that this is an external function (as opposed to an external variable).

The lines following the "main()" function start are enclosed in curly braces. In C terminiolgy, this is called a "body". Whenever you have several lines that go together, you put them in a "body" enclosed with curly braces. Also, note that most of the lines have semicolons at the end. You always end a statement with a semicolon. This does not include compiler statements that start with a "#".

Since the curly braces enclose lines with semicolons, you do not put a semicolon after the last curly brace. This is why the "main()" function has a body with curly braces and no semicolon after the last curly brace, but the "for" loop has a single statement with a semicolon, and no curly braces at all. If you wanted the "for" loop to do more than one thing, you would put all the desired statements in a curl-braces "body".

The next line declares the variable "i" as the integer. Unlike BASIC, where you can invent variables any old time as you need them, you have to declare all the variables a function will use, at the beginning of the function.

C99 is a "small-C" compiler, which means it handles only integer, character, and arrays of integers and characters. This is pretty good though; you can do almost anything you want with this. In C99, an integer is a 16-bit word, and a character is an 8-bit byte.

The next line is a loop. There are several kinds of loops in C, such as the "do...while()", where the statement (or body of statements) are executed until the logical expression in the parentheses of the "while" is no longer true; the "while()" loop, which is the same except that if the expression in the "while" is false to begin with, the body isn't executedat all (the do... while always executes the body at least once; and the "for" loop, which is pretty much the same as a FOR...NEXT loop in BASIC. Inside the parentheses of the "for" are three things - the initial state, a continuation expression, and an increment. The "i++" business is a C peculiarity. It means the same thing as "i=i+1". Sometimes you might see this as "++1", which does the same thing with a subtle difference which isn't important now. If you wanted to increment or change the variable by more than 1, you could write the increment expression any way you wanted, for example, "i=i+3" or "i=i/2" or whatever. Oh, and "i--" (or "--1") will decrement "i" by 1.

As long as the continuation expression of the "for" loop is true, the statement or body following the "for" will be executed. in this case, it si a single statement, which is a call to the external "printf" function with these arguments.

The "printf" function is a powerful and popular C function, and deserves some explanation. The first argument is the for-at string. When the "printf" function is executed the text in the for-at string is printed on the screen. Anywhere a "%" symbol is found, the corresponding argument from the arguments following the format string is printed. The "%d" control causes the argument to be printed as a decimal number. Using "%x" would print it as a hexadecimal number. Using "%c" would convert it to a character (so a number 65 would be printed "A"). and "%s" prints a text string. In this case there is only one "%d", so only one additional argument is expected, which is "i", so "i" will be printed as a decimal number. The "\n" thing is called a "newline", which starts the printing at the beginning of the next line, just like a ";" in a BASIC PRINT command.

C is kind of a cookbook language. Like BASIC, for the most part, you just look up the functions in your manual and figure out what kind of arguments to give it. The same is true in C. In fact, the manual is little else, just the functions and which library and include file you need to use. At least the BASIC manuals tell you how to use the language, which is why so many TIers could get started without any help. I have yet to see a C manual that explains how to use the language, which is why tutorial articles like this one keep getting written.

Ok, a few more things before we compile. I hope you entered the program just as it was shown, with lower case letters. C is a "case sensitive" language, unlike BASIC which automatically converts all the letters to uppercase. And C, unlike BASIC or Pascal, uses lowercase almost everywhere.

The blank lines, spaces, and indentations are not as important. The compiler ignores all that; you could have jumbled everything together on one line, but you can guess how ugly that would be. So the rule is to use spaces or tabs or blank lines or whatever to help your program be understandable. By the way, did you notice the comment? Anything between "/*" and "*/" is ignored by the compiler, so you can explain your code. Or deactivate portions of it for debugging. a very useful attribute of comments!

Okay, let's do this. Run the C99 program out of E/A Option 5. You will be prompted for the input filename (DSK2.TEST_C), an output filename (DSK2.TEST_O), and two options; just press Enter to skip these option for now. The program will chew and convert your C language program file to a TI Assembly language program file. C compiliers for other machines skip this assembly language step, but we have to live with it for now.

If you made any programming mistakes, the compiler will diplay the line with the mistake. The nice thing about C99 is, once you get it to compile with no errors, you are probably home free. If you used TI-Writer to create the program, the compiler will choke on the "tab" line at the end, but you can ignore this.

Now run your assembler, either the TI assembler, or RAG, or whatever you like. This converts the assembly language output of the compiler to object code. Now, load the following library and object file using E/A Option 3:

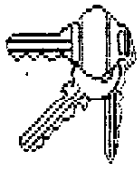
DSK1.CSUP - this is the C99 "kernel". You ALWAYS need this.
DSK1.CFIO - this is the library that goes along with STDIO.
DSK1.PRINTF - this is the external "printf" library file.
DSK2.TEST_O - this is the object file for you program.

Now you can press Enter, and give it the program name "START". If everything worked properly, you will see:

```
Hello number 0
Hello number 1
Hello number 2
Hello number 3
Hello number 4
```

This is a quick introduction to C99, to get you started. Once you get your feet wet, you can look through the manuals of the other libraries, such as for Graphics (do anything XBASIC can do with sprites, character definition, etc., Bitmap mode (write to separate pixel locations on the screen)), Sound, Speech, Math, and other fun fast features of the TI computer.

Feedforth - Jan '96



by W. Leonard Taffs

HAPPY NEW YEAR to everyone!

IN MEMORIAM: JIM PETERSON The anniversary of his passing comes within a few days of this newsletter mailing. It is especially sad for bereaved families to have such memories at what is the most joyous time of the Christian year. Those of us who have suffered similar losses at this time of year, well know what the Peterson family and other bereaved families must be going through. The Peterson family may not know we were thinking of them but it is appropriate to express words of comfort and love, as thanksgiving for the great legacy Jim Peterson left to the TI community. I still sense the pain he must have felt that more of us did not show more interest or take greater advantage of what he had to offer.

I wish I might have met him! Beneath whatever may have been the facade some people perceived him to display, it always seemed to me underneath this was a man who could be very humble and loving. It had to be more than just love of a computer that drove him to do so much for all of us. We are fortunate that in spite of his disappointment at lack of interest shown, discouraging as this must have been, that he kept writing up to the very end. His light still shines for many of us...

These thoughts impelled me to leaf back through some newsletters I borrowed from our library, looking for reprints of anything he had written. The first article I came across was his Tigercub Tips #35, which contained "WORDCOUNT". As I had been asked to prepare this month's D.O.M., I have included this as WORDCOUNTP on your disk. As some schools may be nearing that time when papers are due, this little gem may be useful to students. While copying WORDCOUNT I remembered 2 other such programs that had been published some time ago in MICROpendium. They are modifications of a program by B. Davies of Leander, TX(1986). One program revision, was by Bob Mack (Waynesboro, GA), and the other by Jiri Svoboda of Toronto, Ont. Both these have been added to this month's D.O.M. for those interested in comparing results to Mr. Peterson's program. See the file *READWORDC on this disk.

Using a test file with 50 words the results were:

WORDCOUNTD	(B.Mack rev./"Fast" (Slow version took forever)	61 seconds.
WORDCOUNTJ	(rev. by J. Svoboda)	9 seconds
WORDCOUNTP	(Mr. Jim Peterson)	6 seconds
	with added display:	18 seconds

CHALLENGE. Below is Jim Peterson's "TAKEAWAY" program as found in his TIPS #39. This is an elementary arithmetic program for subtracting. It was part of a continuing series of Math programs he was working on but apparently abandoned when he felt there was insufficient interest shown by the TI community to make it worth his efforts to continue.

When this program is run (makes optional use of the Speech Synthesizer) it shows a face in the upper left portion of the display frame. It is supposed to nod for Yes (correct answer) or shake sideways for No (incorrect answer). Until a correct answer is given the problem remains on the screen.

The challenge is to find if there is an algorithm that could be added to the program (or modification of an existing program algorithm) that would enable it to continue the head nodding and shaking, as this action ceases after the first correct or incorrect answer is given. When you proceed to the next problem, the head no longer responds to right or wrong answers. You have to re-run the program to see the head move again. Here is the program, thanks to Dave Swartz of Rocky Mt. 99ers U.G.!:

```
100 CALL CLEAR :: CALL TITLE
(5,"TAKE AWAY")!by Jim Peter
son
110 DISPLAY AT(3,10):"COPYRI
GHT":TAB(10);"TIGERCUB SOFTW
ARE":TAB(10);"FOR FREE":TAB(
12);" DISTRIBUTION":TAB(11);
"SALE PROHIBITED"
120 CALL PEEK(-28672,A@):: I
F A@=0 THEN 150
130 DATA FINE,NO,GOOD,UHOH,R
IGHT,TRY AGAIN,YES,THAT IS N
OT RIGHT
140 FOR J=1 TO 4 :: READ RIG
HT$(J),WRONG$(J):: NEXT J
150 FOR D=1 TO 1000 :: NEXT
D :: CALL DELSPRITE(ALL)
160 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
170 CALL CHAR(120,"E70042001
8007E0000E700420099423CE7004
20099423C00E700421800BC4200"
)
180 CALL CHAR(124,"0E0004010
0070900700020800QE01000")
190 DISPLAY AT(3,10):"TAKE A
WAY" :: CALL CHAMELEON
```

```
200 CALL COLOR(14,2,2):: CAL
L HCHAR(4,4,143,2):: CALL HC
HAR(5,4,143,2):: CALL SPRITE
(#25,120,11,25,25)
210 T=T+1 :: N=1-(T>5)-(T>15
):: G=10-(T>5)-(T>15)0
 :: H=0-(T>5)-(T>15)
220 X=INT(G*RND+H):: Y=INT(G
*RND+H):: IF Y>X THEN TT=X :
: X=Y :: Y=TT
230 IF X=X2 OR Y=Y2 THEN 220
 :: X2=X :: Y2=Y :: Z=X-Y
240 GOSUB 250 :: GOTO 210
250 GOSUB 260 :: GOSUB 280 :
: GOSUB 310 :: FOR D=1 TO 20
0 :: NEXT D :: CALL DELSPRIT
E(ALL):: DISPLAY AT(18,1)::
CALL CHAMELEON :: CALL SPRIT
E(#24,120,11,25,25):: RETURN
260 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$(Y),J,1)):: NEXT J
270 FOR J=1 TO LEN(STR$(Z)):
: C(J)=VAL(SEG$(STR$(Z),J,1)
):: NEXT J :: W=LEN(STR$(Z))
-LEN(STR$(X)):: RETURN
280 R=96 :: CC=96 :: FOR J=1
 TO N :: CALL SPRITE(#J,48+A
(J),11,R,CC):: CC=CC+16 :: N
EXT J
290 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
300 CALL HCHAR(18,12,95,N*3)
:: CC=CC-16 :: RETURN
310 R=140 :: FOR J=LEN(STR$(
Z))TO 1 STEP -1 :: IF LEN(ST
R$(X))=1 THEN M=CC :: GOTO 3
30
320 FOR M=CC TO CC+8 :: CALL
 LOCATE(#J-W,96,M,#J+4-W,116
,M):: NEXT M
330 IF A(J-W)>=B(J-W)THEN 36
0 :: CALL SPRITE(#28,49,16,9
6,M-9)
340 IF F3=1 THEN 360 :: F1=1
 :: A(J-W-1)=A(J-W-1)-1 :: I
F A(J-W-1)<0 THEN A(J-W-1)=9
 :: F2=1 :: A(J-W-2)=A(J-W-2
)-1
350 CALL SPRITE(#22,48+A(J-W
-1),16,80,M-24):: IF F2=1 TH
EN CALL SPRITE(#21,48+A(J-W-
2),16,80,M-40)
360 CALL SPRITE(#27,45,16,11
6,M-12)
370 CALL SPRITE(#20,63,11,R,
M)
380 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 380
390 CALL DELSPRITE(#20,#28):
: CALL SPRITE(#12+J,K,11,R,M
)
400 IF K-48<>C(J)THEN GOSUB
450 :: CALL DELSPRITE(#12+J)
:: F3=1 :: GOTO 330
410 CALL DELSPRITE(#27):: IF
 F1=1 THEN 420 ELSE IF F2=1
THEN 430 ELSE 440
420 F1=0 :: CALL DELSPRITE(#
J-W-1):: FOR P=80 TO 96 :: C
ALL LOCATE(#22,P,M-24):: NEX
T P :: CALL SPRITE(#J-W-1,48
+A(J-W-1),16,96,M-24):: CALL
 DELSPRITE(#22):: GOTO 440
430 F2=0 :: CALL DELSPRITE(#
J-1-W):: FOR P=80 TO 96 :: C
ALL LOCATE(#21,P,M-24):: NEX
T P :: CALL SPRITE(#J-1-W,48
+A(J-1-W),16,96,M-24):: CALL
 DELSPRITE(#21)
440 CC=CC-16 :: NEXT J :: GO
SUB 480 :: F3=0 :: RETURN
450 DATA 123,124,125,123,124
,125,123,120
460 IF A@=0 THEN 470 :: CALL
 SAY(WRONG$(INT(RND*4+1)))
470 RESTORE 450 :: FOR JJ=1
TO 8 :: READ P :: CALL PATTE
RN(#25,P):: XX=2 250 :: NEXT
 JJ :: RETURN
480 DATA 121,122,121,122,121
,122
490 IF A@=0 THEN 500 :: CALL
 SAY(RIGHT$(INT(4*RND+1)))
500 RESTORE 480 :: FOR JJ=1
TO 6 :: READ P :: CALL PATTE
RN(#25,P):: XX=2 250 :: NEXT
 JJ :: RETURN
510 SUB CHAMELEON
520 M$="1800665AC342DB667E18
8100995AC3A5E78142BD24DB6600
8142992400E7E5AC3A53C241800FF
DB5AFF7EFF00991881006600 18"
```

```
530 RANDOMIZE :: CALL CHAR(1
28,SEG$(M$,INT(43*RND+1)*2-1
,16)):: X=INT(14*RND+3)
540 Y=INT(14*RND+3):: IF Y=X
 THEN 540 :: CALL COLOR(13,X
,Y)
550 CALL HCHAR(1,2,128,30)::
 CALL HCHAR(24,2,128,30):: C
ALL VCHAR(1,31,128,96):: SUB
END
560 SUB TITLE(S,T$)
570 CALL SCREEN(S):: L=LEN(T
$):: CALL MAGNIFY(2)
580 FOR J=1 TO L :: CALL SPR
ITE(#J,ASC(SEG$(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
590 SUBEND
```

It is curious what Mr. Peterson intended to DISPLAY in line 250 where "DISPLAY AT(10,1)" appears.

GLITCH found. In WINDOW:1-6 which appears on this disk of the month, there is a minor glitch when you attempt, after opening 6 files, to use the option of entering "7" at the "Print Which Window?" prompt. When records are displayed in this program, you have the option of printing any record display from any open files, by pressing the respective window number. Not shown in on-screen help is the fact that you can, when pressing "P" to print a window, press "7" to print all resident records being displayed at the time. I discovered (too late to correct before this disk of the month was done) that when you press "7" it will print all displayed records but, with my STAR NX-10 printer, it will overstrike a few characters, at least in the first 2 records. I will attempt to put the correction for this in next month's column. If any of you beat me to this correction, I would appreciate your sharing it with me.

HALTING A PROGRAM. The CALL KEY routine, of which two Extended Basic versions are shown here, can be versatile methods of interrupting or halting a program. (The same routines can be used in TI Basic but require quite different program lines).

This version halts the program immediately when the program comes to this line:

```
1) 100 CALL KEY(0,K,S):: IF S<1 THEN 100
       (Pressing ANY key will continue program)
```

This version, by contrast, will not halt the program until you press a key:

```
2) 100 CALL KEY(0,K,S):: IF S<>1 THEN 130
    110 IF K=? THEN (line number or program action to
        be taken if this key is pressed)
   120 CALL KEY(0,K,S):: IF S<>1 THEN 120
       (Pressing ANY key will halt program. Then,
        pressing ANY key a second time will
        continue the program, but NOTE exceptions
        in following notes)
   130 (Program lines continued)
```

Both above CALL KEY versions will allow you to continue, once they have halted the program, by pressing ANY key. EXCEPTIONS to this are if there are any other conditional (defined keys) specified and inserted between the full CALL KEY routine.

In the case of the first example, conditions/keys can be inserted between the line separator (::) and "IF S<1 THEN 100".

In the second example, conditional key statements can be inserted between line numbers 100 and 120. Line 110 is a condition specifier. It will direct the computer to do a certain thing when the specified key is pressed; otherwise pressing any OTHER key will continue the program.

The question mark in line 110 is the position to put the ASCII number of the key to be pressed, a choice up to the programmer. This makes it possible to escape from a running program. Quite commonly, the upper case letter "Q" is used to QUIT a program. The ASCII number for "Q" is 81. Line 110 can be: IF K=81 THEN (line number direction to quitting portion of the program). Be certain, if any files have been opened by the program, that the program is directed so as not to bypass any file CLOSE statement(s). A safeguard version in line 110 (for any version of of CALL KEY using "0" as the first digit in its parenthesis), is to include both upper case and lower case ASCII numbers. For using "Q" to quit, 113 is the ASCII number for "q":

```
110 IF (K=81)+(K=113) THEN (to program ending)
```

Between lines 100 and 120, other program line direction options can be included in addition to line 110, such as if K=80 then GOTO a Printer option (80="P") or if K=82 then go back to a portion of the program you wish to repeat (82="R"), etc. Where a program is directed back to any repeat action, program running conditions must be understood. If program is to re-read DATA, a RESTORE statement is important, or if it is to re-read a file, the file must first be closed, and then re-opened. Flags and counters in the program may need to be cleared or re-set! These are some brief examples of other CALL KEY options.

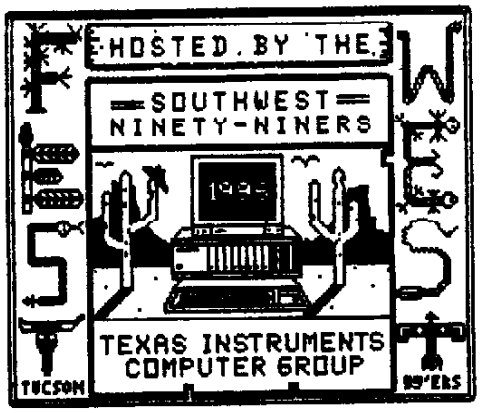
ON EDITING ONE'S WRITING. Our newsletter editor, BJ Mathis, in discussing the articles people submit for our newsletter, gave some good advice. She said: "Look through your article to see how often you use the word 'that'--see if it's really necessary to use it!". Remembering that (OOPS!) advice, I checked the copy of a letter I had just mailed. The word "that" appeared 31 times. Only five uses of the word were necessary for proper syntax!

Let us remember Jim Peterson and his family in our prayers while we wish each other HAPPY NEW YEAR!

-- A N N O U N C I N G --

TI Fest West '96

A Texas Instruments 99/4A
& MYARC Geneve 9640
Computer Users Convention



Fri Feb 16, 1996 - Set-Up
Sat Feb 17, 1996 - 9am to 6pm
Sun Feb 18, 1996 - 9am to Noon - Pack-Up

FEST WEST '96 IS THE BEST IN THE WEST!! Fest West, in past years, has been held in Salt Lake City, Phoenix, Anaheim, Tucson, San Diego, Las Vegas, and Los Angeles. This year, in what promises to be the best Fest West yet, it will be held again in Tucson, Arizona and hosted by the SouthWest Ninety Niners.

FEST WEST '96 will have a new format!! Vendors will Set-Up Friday evening, and we will pre-register as many people as we can, at that time. Saturday will be the main Fest West event, with door prizes and raffle prizes given at the end of the day. The Hospitality Room will be available for the usual "Fest West gab sessions" both Friday and Saturday evening as late as we want it. Vendors will not have to pack-up until Sunday morning, so they can join the "gab sessions"!!

Dealers, vendors, and user groups from all over the United States will be present, offering a complete selection of hardware, software, & accessories for the TI-99/4A and the Myarc Geneve 9640 computers.

Ramada Inn University, 1601 N. Oracle Road. Tucson. is giving us great rates at the peak of our tourist season. Singles or doubles will be only $55 per night. For reservations call, 1-800-777-2999, tell them you're attending Fest West '96.

Ramada Inn University is providing airport/hotel van shuttle at a charge of $5 per person round trip. This shuttle service requires a minimum 48 hours prior notice, so be sure to reserve the shuttle service when you make your hotel reservations. This is NOT the same hotel we had in 1990 and 1994.

Besides attending Fest West '96, Tucson is a great place to visit in February. With normally sunny days, warm temperatures, and beautiful evenings, Tucson is a very active community during the winter months. After Fest West '96, consider a vacation in the "Old Pueblo". Things to see in and around Tucson include historic downtown Tucson, The Arizona-Sonora Desert Museum (one of the 10 best zoos in the country, plus a museum and botanical gardens), Reid Park Zoo, the Mission San Xavier Del Bac, the Titan Missile Museum, Pima Air Museum (a huge collection of historic airplanes), and many, many more sights to see!

For more information about TI Fest West '96
or SouthWest Ninety-Niners User Group
PO Box 17831 - Tucson, AZ 85731
Call BJ Mathis - (520)747-5046 or Tom Wills (520)886-2460
Or Cactus Patch BBS (520)290-6277

Admission $4 - 15 & under free when accompanied by adult

TI Fest West '96 - Best in the West!

Wise Buys

The following information is provided as a service to our members. The items listed are for sale by the individuals indicated and are subject to prior sale. The group assumes no responsibility for items listed and makes no claims as to their condition or interface compatibility with the TI-99/4A computer. Only computer related items will be accepted for publication in this newsletter.

$75 - 14" Commodore Color Monitor
Model 1720

Ed Chase (520)299-6511 or #118 on CP

$50 TI RS-232 card o.b.o.
$8 Speech Synthesizer
$5 TI Color monitor cable
$15 IBM Style Keyboard
$2 Continuous Form Feed Legal Envelopes

$1 Disk Manager 2
$1 Football
$1 Household Budget Management
$10 Mini Memory
$1 Munchman
$2 Parsec
$1 Personal Real Estate
$1 Personal Record Keeping
$1 Personal Report Generator
$1 Terminal Emulator II
$1 The Attack
$30 TI Extended Basic
$1 Video Chess
$1 Companion Word Processor
$1 Hypercopy
$1 Pre-Scan It
$1 Quality Soft Screen Dump
$1 Beginner's Basic Tutor
$1 Teach Yourself Basic

Tom Wills (520)886-2460 or #1 on CP

The Altman Fairware List
$2 each postage included - 1995 edition

Modules

$1 Adventure (module only)
$3 Editor Assembler
$1 Home Financial Decisions
$1 Household Budget Management
$5 Multiplan
$2 Parsec
$1 Personal Real Estate
$1 Personal Record Keeping
$1 Personal Report Generator
$3 Pole Position
$1 Securities Analysis
$10 Speech Editor
$1 Tax Investment Record Keeping
$2 TI Writer
$1 Tunnels of Doom (module only)

Disk Programs
$10 Backsteine

Printer Ribbons

$2 Epson MX-80 $4.50 MX-1020
$3 MX-10 or MX-1000 $3.50 MX-2400

Cassette Programs
$1 Teach Yourself Basic
$1 TI-99/4A Trivia Data Base

BJ Mathis(520)747-5046 #3 on CP

Books
$3 Basic Programs for the Home
$2 Computer Playground
$10.50 Home Publishing on the TI-99/4A
Supplement #3 w/disk
$2 Programs for the TI Home Computer
$3 The Word Processing Book
$3 The Writers by Harry Brashear
$2 Using & Programming the TI-99/4A

$10 TI 99-4/A Console
TI Keyboard Overlays $3ea or 5/$10
$10 Speech Synthesizer
$10 Widget
Joysticks (several pair) - make offer
$10 Thermal Serial Printer (Crackd Cvr)
$60 o.b.o. P-Code Card w/documentation
$4 Replacement Console Power Supplies
$3 Joystick Adapter
$10 Cassette Player/Recorder
$2 Replacement Keyboard
$2 Cassette Cable
$8 PIO Printer Cable
$8 Modem Cable (Telco Ready)
$2 RS-232 y-cable
50/$1 Disk Labels
500/$1 Mailing Labels
Tractor feed 1-across

BJ Mathis(520)747-5046 #3 on CP

Super AMS Order Form

Name _____ Date _____

Address _____ Phone _____

[ ] Average User [ ] Programmer [ ] Hardware Designer

Product Name	Price Each	No.Items	Total
Super AMS 128K	$85.00		
Super AMS 256K	$100.00		
Shipping & Handling	$5.00		
	TOTAL PRICE		

[ ] Cash [ ] Check (#) [ ] Money Order (#)

Please make check or money order payable to SouthWest Ninety-Niners
Send Order to: SouthWest Ninety-Niners, PO Box 17831, Tucson, AZ 85731

SouthWest Ninety-Niners Nomination Form

Indicate your nominations for the offices listed below, cut and mail this form to the group P.O. Box, hand it to a nominating committee member, or call a nominating committee member. You may nominate yourself or any other member in good standing. Elections will be held at the January meeting.

Nominating Committee Members are Mike Doane - 298-3835, Tom Wills - 886-2460, and Tom Holder - 299-6366.

President _____ Secretary _____

Vice-President _____ Treasurer _____

Membership Report

Rolf & Harriet Kradenpoth, Earl Colvin, Rod Stallard, Martin Marcinko, and Robert Zink renewed SW99ers memberships during the month of December. We will mail newsletters to 44 members, plus 33 groups and vendors this month.

General Users Workshop

Canceled for a while.

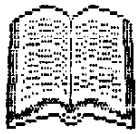
Members Meet

THURSDAY, January 4th at VFW post 4903. Election of SW99ers officers for 1996. SW99ers Disk Library 1/2 price sale. BJ will have lots of "new" used items for sale (some free!).

Exec Meeting

To be set up by new executive committee.

Contents



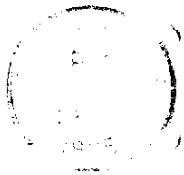
Who Do Ya' Call?

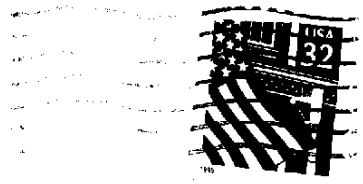
Richard Baron - Disk Librarian	885-4812
Mike Doane - Treas/Technical Assist	298-3835
BJ Mathis - Pres/Editor/Library Chmn	747-5046
Jack Mathis - Technical Assist	747-5046
Matt Matthews - Lending/News Lib	428-6910
Les Neff - Secretary	327-6437
Tom Wills - VP/BBS SysOp/Lending Lib	886-2460
Open - DOM Preparation	??????
Cactus Patch BBS	290-6277

(Area code 520 for phone numbers above)

February Newsletter Deadline))January 19th, 1996((
)))) Note! Early Deadline! ((((

SouthWest Ninety-Niners
PO Box 17831
Tucson, AZ 85731





Dallas TI HC Group 9512
Dallas 99 Interface
PO Box 29863
Dallas TX
75229

75229-0863