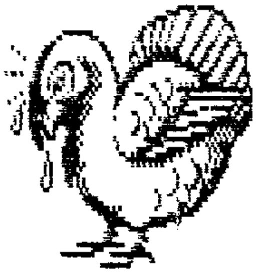
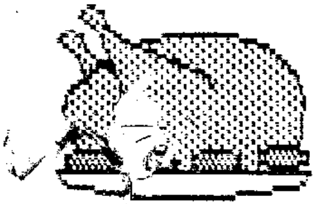
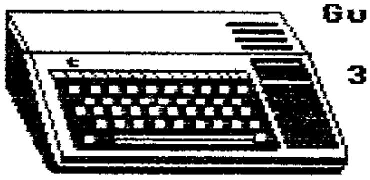
GUILFORD 99'ERS NEWSLETTER





SUPPORTING THE TEXAS INSTRUMENTS TI-99/4A COMPUTER



GUILFORD 99'ERS UG
3202 CANTERBURY DR
GREENSBORO NC
27408

Wood Duck
29 USA

TO:

Bob Carmany
1504 Larson Street
Greensboro, NC 27407

VOLUME 8 NUMBER 11 NOVEMBER 1991

George von Seth, Pres. (292-2035)   Bob Carmany, Newsletter Ed (855-1538)
Tony Kleen, Sec/Treas (924-6344)   Bill Woodruff,Pgm/Library (228-1892)
BBS: (919)621-2623 --ROS
+-
The Guilford 99'er Users' Group Newsletter is free to dues paying members (One copy per family, please). Dues are $12.00 per family, per year. Send check to:Tony Kleen c/o 3202 Canterbury Dr., Greensboro, NC 27408. The Software Library is for dues paying members only. (Bob Carmany Ed)
+-

OUR NEXT MEETING

DATE: November 12,1991 Time: 7:30 PM. Place: Glenwood Recreation Center, 2010 S. Chapman Street.

Program for this meeting will be a surprise -- both to the members of the Users' Group and to Tony Kleen who will present it. Tony always manages to come up with something interesting and this should be no exception. Be sure to be there!!

EDITOR'S NOTES

Once again, we are faced with the end of the year and another decision about where to go with our Users' Group. The fact of the matter is that our meager membership is steadily shrinking as the months go on. Submissions to the newsletter have sunk to zero with the exception of Tony Kleen's articles and there doesn't appear to be any prospects for anything substantive in the immediate future. Once, I had a library of articles from newsletters written in the heyday of our machine but even those have been used at least once. So, here we are!

Another factor to consider is that we will need to have a new Executive Officer, a new Secretary/Treasurer, and a new Newsletter Editor selected by year's end. There doesn't seem to be a wealth of volunteers! I know that I would like to take a rest for awhile and let someone else handle the newsletter for the coming year ---any takers?

A decision of some sort about the future of the Users' Group will have to be made at the December meeting at the outside.

SPELLCHECKERS

By Bob Carmany

There are currently two spelling checker programs for the TI. The first is SPELLCHECK by Dragonslayer Software and the second is the recent etry from Asgard Software named SPELLIT. Let's compare them to each other.

SPELLCHECK

1) It comprises about 500 disk sectors which includes the program itself and two main dictionaries. There is a seperate XB program to create the "seed" for additional dictionaries.

2) SPELLCHECK is easily configurable with a disk sector editor (and a little common sense) to run from any disk drive --either a physical drive or a RAMdisk.

3) It lacks the facility to run in 80-columns.

4) It is slightly slower than SPELLIT.

5) You can go directly from the Funnelweb Editor to SPELLCHECK to edit a document and it installs quite easily as a F'WEB menu option.

6) It requires the "fixed" document to be reformatted if the corrected words are not the same size as those they replace.

7) The destination drive number and filename are flexible and require entry by the user. They can go to any drive or RAMdisk and have any filename.

8) When new words are added to the user-created dictionaries, they are automatically alphabetized before the lot is saved. You can select and name any number of dictionaries for special applications. This makes it easy to work with the dictionaries and modify them.

9) There is no facility for synonyms.

SPELLIT

1) SPELLIT requires 850+ sectors for the program and the included dictionaries. The minimum reasonable requirement (without disk switching) is a DSDD drive or a RAMdisk.

2) As delivered, the program will not access a RAMdisk above >1100. That creates problems for those with multiple RAMdisks are RAMdisks with drives installed above that location. To be fair, there is a fix from Ron Kleinschafer to solve THAT problem.

3) It operates in either 40 or 80-column mode.

4) It is a bit faster than SPELLCHECK.

5) There is no direct entry allowed from F'WEB. A memory contention causes the mailbox to be 'trashed' and you have to leave F'WEB completely to check your document.

6) Corrected words are automatically replaced when the document is re-written.

7) Corrected files are automatically saved to the same disk as the original. There is no facility to use an alternative drive or filename.

8) All added words are saved to a dictionary named 'USER' and are placed at the end of the dictionary in no special order. This limits the user to a single supplementary dictionary.

9) A (L)ookup feature presents possible spelling variants from the main dictionaries.

Both programs have dictionaries that are extensible with the addition of new words not found in the main dictionaries. SPELLIT seems to be restricted to a single supplementary dictionary whereas SPELLCHECK can accept an unlimited number of 2,000 word supplementary dictionaries.

When using SPELLIT, you will have to add the new words to the main dictionaries for speed and ease of use.

SPELLCHECK views upper and lowercase words in a document as different until they are entered into a supplementary dictionary. An advantage is the use of an XB program to create the basis for a dictionary of supplementary words.

In trying to use SPELLIT with my twin Quests, I ran into so many problems that I finally decided to switch back to the Dragonslayer program. It runs fine from a RAMdisk.

Symmetrical Redefined Characters

by Jim Peterson

This tutorial will help teach you how to create random symmetrical redefined characters. I doubt that I'm the first one who ever thought of this, but I've never seen it in anyone else's programs and I don't think it can be done in BASIC on any computer other than the TI.

In its basic form it goes like this:

```
100 DIM A$(16)
```

```
110 DATA 00,18,24,3C,42,5A,66,7E,81,99,A5,BD,C3,DB,E7,FF
120 FOR J=1 TO 16
130 READ A$(J)
140 NEXT J
150 FOR L=1 TO 4
160 RANDOMIZE
170 X=INT(16*RND+1)
180 B$=B$&A$(X)
190 C$=A$(X)&C$
200 NEXT L
210 CALL CHAR(65,B$&C$)
```

Now, no one should ever use a routine without understanding it, because you won't be able to debug it and you won't be able to modify it. So, let's go through this. If you take a good look at the chart on page 109 of your "BEGINNER'S BASIC", or page II-77 of the "USERS REFERENCE GUIDE", you will see that those pairs of hexadecimal numbers in line 110 represent rows of bits which are mirror images of each other. Therefore, if we assemble a character from these pairs, it will have left-right symmetry. So, lines 100-140 read these pairs into an array. Then, lines 120-200 go through a loop four (4) times, each time picking one of these pairs at random, by randomly picking a subscript number between 1 and 16. The top half of the hex code of our redefined character is built up by adding these pairs to the end of string B$, which starts out as a blank. For instance, if on the four (4) loops the random values generated for X are 1, 2, 3 and 4, the pairs selected are 00, 18, 24 and 3C, and B$ is successively built up as 00, 0018, 001824, and finally 0018243C. At the same time, C$ is built up with the same pairs in reverse order, as 00, 1800, 241800, 3C241800. Finally line 210 redefines ASCII character 65 as being string B$&C$, which is "0018243C3C241800", and which is symmetrical top-to-bottom as well as left-to-right.

Now that you understand how it works, let's program it a bit more efficiently.

```
100 FOR L=1 TO 4
110 RANDOMIZE
120 X$=SEG$("0018243C425A667E8199 A5BDC3DBE7FF",INT(16*RND+1)*2-1,2)
130 B$=B$&X$
140 C$=X$&C$
150 NEXT L
160 CALL CHAR(65,B$&C$)
```

You can cram that into one line of Extended Basic! So, what's it good for? Well, [illegible]t's add:

```
90 FOR CH=40 TO 152 STEP 8
```

change 160 to CALL CHAR(CH,B$&C$)

```
170 B$=""

180 C$=""

190 NEXT CH
```

Now we've redefined the first character of sets 2 through 16. Don't forget lines 170 and 180. Since B$ and C$ are formed by adding onto themselves, they must be cancelled out before we start over or they will just keep on adding onto themselves. Next, let's give each character set a foreground color and a different background color.

```
200 FOR SET=2 TO 16

210 X=INT(15*RND+2)

220 Y=INT(15*RND+2)

230 IF Y=X THEN 220

[illegible]0 CALL COLOR(SET,X,Y)

250 NEXT SET
```

Now for the fun...

```
80 CALL CLEAR

260 CALL SCREEN(5)

270 CALL HCHAR(INT(24*RND+1), INT(32*RND+1),INT(15*RND+1)*8 +32,INT(10*RND+1))

280 CALL VCHAR(INT(24*RND+1), INT(32*RND+1),INT(15*RND+1)*8 +32,INT(10*RND+1))

290 IF INT(10*RND)<>0 THEN 270

300 CALL CLEAR

310 GOTO 270
```

Or if you're in Extended Basic, let's change:

```
90 FOR CH=40 TO 136 STEP 4

155 SP=SP+1

[illegible]0 CALL CHAR(CH,RPT$(B$&C$,4))

165   CALL   SPRITE(#SP,CH,INT(15*RND+2),1,1,    INT(10*RND+2)-INT(10*RND+2),INT
(10*RND+2)-INT(10*RND+2))

85 CALL SCREEN(5)
```

```
200 FOR D=1 TO 100

210 NEXT D

220 CALL MAGNIFY(INT(4*RND+1))

230 GOTO 200
```

and delete 240-310.

You would prefer something a bit more useful? OK, let's try a different variation of the same principle.

```
100 CALL CLEAR

110 RANDOMIZE
120 DATA TIGERCUB PRESENTS,THE ,CHAMELEON,SCREEN BORDER,AND,WIPE ,by Jim
Peterson," "," TOUCH ANY KEY"

130 M$="1800665AC342DB667E1881009 95AC3A5E78142DB24BD66008142992400
7E5AC3A5C3241800FFDB5AFF7EFF00991 88100660018"

140 RESTORE 120

150 FOR P=1 TO 9

160 READ A$

170 PRINT TAB(15-LEN(A$)/2);A$;" "

180 NEXT P

185 PRINT :::::

190 GOSUB 240

200 CALL KEY(0,K,ST)

210 IF ST=0 THEN 200

220 GOSUB 320

230 GOTO 140

240 CALL CHAR(128,SEG$(M$,INT(43* RND+1)*2-1,16))

250 X=INT(15*RND+2)

260 Y=INT(15*RND+2)

270 IF Y=X THEN 260

280 CALL COLOR(13,X,Y)

290 CALL HCHAR(24,1,128,64)

300 CALL VCHAR(1,31,128,96)
```

```
0 RETURN

320 T=T+1-ABS(T=2)*2

330 ON T GOTO 340,360

340 CALL VCHAR(1,3,128,768)

350 GOTO 370

360 CALL HCHAR(1,1,128,768)

370 CALL CLEAR

380 RETURN
```

In this one, M$ consists of any of those symmetrical pairs typed in at random, and we define a character which has only left/right symmetry by randomly pulling out any sequence of 16 of these.

Now start tapping any key until you find an appropriately evil-looking alien space ship or man-eating bug for your game program.

If you consult that chart in "BEGINNER'S BASIC" again, you will find that the first eight (8) of those pairs do not turn on the first or last bit, therefore do not fill a print space. So, let's enter another program:

```
100 CALL CLEAR

110 FOR CH=129 TO 154

120 RANDOMIZE

130 FOR L=1 TO 3

140 X$=SEG$("0018243C425A667E",INT(8 *RND+1)*2-1,2)

150 B$=B$&X$

160 C$=X$&C$

170 NEXT L

180 CALL CHAR(CH,"00"&B$&C$)

190 B$=""

200 C$=""

210 NEXT CH

0 INPUT M$

230 GOTO 220
```

Now type any of the letters with the CTRL key held down - your computer has a built-in Venusian alphabet!

Many other effects and variations are possible. I use this routine frequently in my Tigercub programs. It provides the infinite variety of kaleidosco displays in Kaleidovision, Multivision and Ten Thousand Sights. It enables me to provide a completely different assortment of colorful cards to turn over in each new game of Match a Patch, and a new screenfull of walls in each game of Getaway. It provides the colorful characters of the Mongolian Typewriter for the little tots, and the rainbow displays that reward correct answers in Kinderminus, and many more. My Color Programming Tutor will show you ways to make the displays even more colorful, and my Random Character Generator will give you your choice of 8 different types of these characters, print out an assortment of 40 of them, display them singly and in strips and blocks and as a sprite, allow you to change their foreground and background colors, print out their hex code, and enlarge them to your choice of 3 sizes, even filling the screen.

I have sad news for our members. Our much respected Director, George von Seth, passed away October 29th. We all treasured George's warmth and sincerity. We grieve with the family for our great loss. May God bless ya', George.