QB-MONITOR
QB-99'ers U.G. NEWSLETTER

### OCTOBER

The QB 99'ers meets the second Saturday of each month September through May, at Queensborough Community College, Bayside New York, room S225, at 2 P.M. Calendar at right shows dates

October						1989
S	M	T	W	T	F	S
1	2	3	4	5	6	7
8	9	10	11	12	13	(14)
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31				

November						1989
S	M	T	W	T	F	S
			1	2	3	4
5	6	7	8	9	10	(11)
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30		

GET EVEN WITH THE U.S.P.S.
WRITE LOTS OF LETTERS!
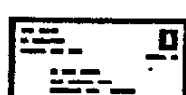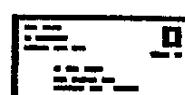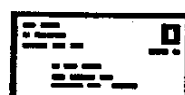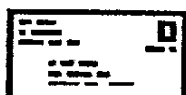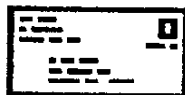
LETTERS – LETTERS – LETTERS
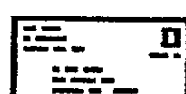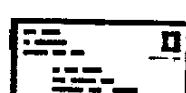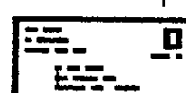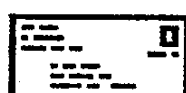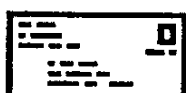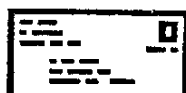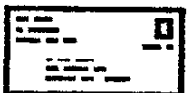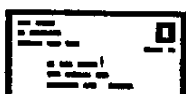
```
      ------
 --------!----!---------
/ TI-WRITER TOOLBOX #6  /  \
/_____/ QB \
!     by Ed Machonis    !99ers!
!_____!_____!
```

It has been a year since we looked into the tool box, mostly because after 5 articles only two people would admit to having read any of them and one was a member of one of our exchange groups.

Despite this underwhelming demand, I was tempted to open the tool box again by a recent article in one of our exchange newsletters on addressing envelopes. I was struck by the similarity to the method I use, even to the point of using the same file name. However, since I do use a few gimmicks not described in the exchange article, I decided to throw in my own two cents.

Before we get into addressing envelopes, it may be wise to generate something to put in those envelopes. To a lot of people, unless they have a long and important letter to write, writing a letter with TI-Writer seems to be more bother that it's worth. The trouble is they let the medium get in the way of the message. They are more concerned with the mechanics of producing the letter than what they want to say.

Some will prefer to use their typewriters - just roll in a piece of paper and type away - others will resort to (for shame) a piece of paper and some purple ink. But, believe it or not, that TI-99 can really make the job simple. All it takes is a system and a little practice.

I carry on a rather extensive correspondence which would not be possible without TI-Writer, especially with my typing. I try to keep most of my letters short, a couple of paragraphs or so. Putting these on a full sheet of paper seems a bit ridiculous and invites one to ramble on and on. I have found that by cutting a standard sheet of paper in half, the resulting 5-1/2 x 8-1/2 pieces are just about right for the average letter. (You can also obtain 6" x 9" blank pads at stationery and variety stores; even some super-markets carry them.)

Allowing for a 3/4" margin on each side, leaves 4" for the text. Hey, that's 40 columns, just the width of my screen. I can load the Editor, set my margins at 0 and 40, turn off the line numbers, type out my letter, print it from the Editor and I'm done! What You See Is What You Get. No windowing, no formatting, no bouncing between the Editor and Formatter to make changes. If a long word leaves part of the preceding line blank, it is easy to hyphenate the word so it breaks over two lines. (80 Columns? Who needs it? People with good monitors and good eyesight. The screen doesn't get any wider - the characters get smaller!)

But that ain't all folks. With just a little one time effort you can pretty up your letters something fierce. How about having your name and address automatically printed at the top of the page? In expanded compressed double strike type style? Here' how.

We will create a file with the necessary print codes and our name address which we will load in before we write our letter. I say before only because it will also contain our screen margins and save that step.

Load in the Editor, enter T for Tabs and set the R for Right Margin at 40; The L should be at 0. (I know this adds up to 41 columns, but Word Wrap will cut it down to 40.) If you like your paragraphs indented, set I at column 3. Pressing Enter should put you on Line 1, where we will enter our Print codes. We shall use the Special Character Mode to do this.

The Special Character Mode enables you to send ASCII character codes 0-31 to your printer. You enter and leave the Special Character Mode by pressing CONTROL U. The ASCII codes, required key presses and the screen displays obtained in this mode are described on page 146 of the TI-Writer manual.

The code that especially interests us is ESCAPE or CHR$(27). Escape is used to tell the printer that the next character is a printer control code and not to print it. If the printer receives a Capital "E" it will print it, but if Escape precedes that E, it will change the type style to Emphasized and the E will not be printed.

TI-Writer Toolbox #6 Cont'd........Page 2

Escape is obtained by pressing CONTROL U then FUNCTION R. Pressing CONTROL U again will enable us to leave the special character mode. I will use this symbol "¦" to represent this series of key presses. We shall also need to enter CHR$(15) for compressed type style and it is obtained by pressing CONTROL U then SHIFT O (NOT Zero!) and another CONTROL U to leave special character mode. The symbol "˜" will represent CHR$(15).

Enter the following print codes on Line 1.

¦@¦G¦W1˜

The required key presses are: (The slant, "/", is used as a separator – Don't press it.) CONTROL U / SHIFT R / CONTROL U / @ / CONTROL U / SHIFT R / CONTROL U / G / CONTROL U / SHIFT R / CONTROL U / W / 1 / CONTROL U / SHIFT O / CONTROL U

These represent Master Reset (¦@), Double Strike (¦G), Expanded (¦W1) and Compressed (˜). This is my favorite type style for addresses and we shall meet it again.

On the next three lines, starting about column 15, enter your name and address. (If desired, your phone number can be entered on an additional line.) Put one of your half sheets of paper in the printer and print the file from the command mode. Nice, huh? Next we must center the three lines of text on your paper. This is best done by trial and error. Turn off Word Wrap and insert or delete spaces at the beginning of each line until they are centered when you print them.

Turn the Word Wrap back on, skip a line and enter the following print code to cancel our previous codes, select Emphasized type and set the Left Margin at column 5:

¦@¦E¦1{

"{" represents CONTROL U / SHIFT E / CONTROL U and is the equivalent of CHR$(5). The next to last character is a lower case letter "L", not the figure one. These codes work for my Epson RX 80 and should work for most Epson compatibles.

Enter a blank line then save the file (SF) under the name PERSONAL.

Now when you have a letter to write, just load (LF) PERSONAL, turn off the line numbers and write your letter. Put the date on the last line of the file you have just loaded. When your letter is finished, save it under a unique file name and print it from the command line with PF. DONE, NEXT CASE!

If you feel that computer printed letters are too impersonal for personal correspondence, you can try printing in Italics, which appears closer to Script. Just add ¦4 at the end of the printer codes on Line 5.

Paper 8-1/2" long can hold 51 lines. Allowing 3 lines for top and bottom margins, this leaves 45 lines for header, date, salutation, close and signature. Turning the Line Numbers back on will show you how it fits. If you letter is a little over, not to worry. Merely imbed a print code (¦1) on the first line to decrease the line spacing. At 7 lines per inch, the paper will hold a total of 60 lines.

If you find you are over 55 lines, no problem. After typing PF, when the printer default appears press FUNCTION 2, Insert Character, then type: 1 <space> 45 (or whatever) <space> ENTER. Lines 1 to 45 will be printed. Insert a new sheet of paper and command PF again. The printer default will now include 1 <space> 45 <space> before the printer name. Type over these with: 46 <space> E <space> and the rest of the letter will be printed.

But there will be times you will want to ramble on and a full sheet is called for. I use the same type of header on the standard sheet. You've already done most of the work. Go to the command line and select T for Tabs. If you put an I on 3, put one on 6. Cursor over to column 70 and type R. No need to blank out the old ones, Tex knows what you want. Press Enter.

TI-Writer Toolbox #6 Cont'd.......Page 3

Now we must center our name and address on a full size sheet. Turn off Word Wrap and, on Lines 2 thru 4, insert about 20 spaces before your name and address. Test print and add or delete spaces until you have the lines centered. Turn Word Wrap back on and save the file under the name HEADER. When using this file, if you are going to print directly from the Editor, you will of course (curse?) have to put up with the windowing. If you will be printing through the Formatter, change the TABS to L=0 and R=40.

Now we finally get to addressing envelopes and you will see why I insisted on putting the horse before the cart. Still with this same file, go to the first line and at the end of the printer codes add !8. (Remember "!" represents CONTROL U / FUNCTION R / CONTROL U.) This code will turn the End of Paper Sensor OFF.

On the next three lines, delete all those leading blank spaces before your name and address so it is flush to the left margin. This is now our return address. (If you included a phone number, delete it.) Go to the next line and insert about 6 blank lines so that our Left Margin Print Code is on Line 12. Go to Line 12 and put your cursor on that funny looking 5 and type ! (Exclamation Point - Don't use CONTROL U). This changes the left margin to column 33. (33 is the ASCII Code for !. In Expanded Compressed Style Column 33 is about the mid-point of a legal size envelope.)

Add about 5 blank lines after this code. This next step is optional, it turns the End of Paper Sensor back ON, but it's also a bit of a pain. On that last line (17) enter !@. When this code is sent to the printer, it will cancel all prior codes, the printer will realize it's out of paper, take itself Off Line and sound the alarm beeper if you don't have it turned off with the dip switch. If you are sure you will turn the printer off and on before using it for purposes other than addressing envelopes you can skip this code. I use it, I don't like printing on a bare platen.

Now go to TABS and set the Right Margin at Column 34. This will help prevent entering a long address line which would break over two lines on the

envelope. I save this file under the name ENVELOPE, you may want to use ENVLEGAL.

When you want to address an envelope, load the envelope into the printer, flap open, type the addressee's name and address on lines 13 through 16, flush with the left margin, command PF and there you are. On the RX 80, I find it easiest to align the envelope horizontally by aligning the bottom of the envelope one quarter inch above the end of the plastic case behind the platen.

Often you will not even have to type in the name and address. You may have already typed it in the letter. In these cases, note the Line Numbers the address is on so that they can be merged into your ENVELOPE file. Say the address is on Lines 8 thru 10. When you are done printing your letter, load in the ENVELOPE file.

Go to the Command Line and type LF. At the file name prompt, cursor to the beginning of the line and type 12 (The line number in your ENVELOPE file after which the address is to be merged) <space> 8 (The first line of the address in your letter file) <space> 10 (the last line of the address) <space> DSKn.LETTERFILE. Viola! The address from your letter file has been copied into ENVELOPE. Just command PF and your envelope is addressed.

Of course, those 5-1/2 x 8-1/2 inch sheets can get lost inside of a legal size envelope, so we will modify our file for the smaller personal size envelope.

Select TABS and set the Right Margin at 25 to prevent too long an address line from being entered. Go to to Line 12 and place your cursor over the Exclamation Point. Press CONTROL U / SHIFT Y / CONTROL U. This will set the Left margin at Column 25, the equivalent of Column 30 in Pica. I save this file under the name ENVPERSON.

These instructions are a bit wordy and therefore look complicated. That's only because I have tried to make them Idiot Proof. Try it, you'll find the process rather simple, and you only have to do the job once. After that, the files will be working for you. As my wife had been heard to remark, "There's No Efficiency Expert Like A Lazy Man."

QB MONITOR ~ QB-99'er NEWSLETTER
Reprinted From JOHNSON SPACE CENTER U.G. Newsletter
IDIOT PROOF FORM LETTERS
by John Owen, JSC UG-Aug., 1989

I always wanted to write FORM LETTERS with my TI99/4 but got tangled up in the instructions that TI provided. The following is an IDIOT PROOF PROCEDURE that I can understand and remember. KIS (Keep It Simple)-FOLLOW STEPS 1,2,3.

1. PREPARE A FORM LETTER

Write your form letter and put in *n* where you expect to take data from your VALUE file to fill in the *socket*. See Example #1. Note that this FORM LETTER is set up to FOLD, STAMP and MAIL (no envelope required). Note that anywhere the form letter has a "*n*", the Formatter will automatically get the data needed to fill this "socket" from the matching "number" in the VALUE file.

Add Formatter Codes (as shown) to your form letter. The Formatter does all the formatting. DO NOT put Formatter Codes in the VALUE file

Save this letter under any filename (i.e : DSK1.FORM-LETR)

2. PREPARE A "VALUE" (NAMES/DATA)FILE

Use the *NUMBERS* generated in Example #1 FORM-LETTER to prepare a VALUE file as shown in Example #2.

This VALUE file has seven line enteries for each NAMES/DATA SET. There is one NAMES/DATA SET for each person that owes dues.

Note: There is NO "*" before line #1 for the first set of NAMES/DATA in the VALUE file. There is an "*" between and after each DATA SET in the NAMES/DATA file.

I only show three DATA SETS in this VALUE file. There could be humdreds of NAMES/DATA SETS in this file and the Formatter will print a FORM LETTER for each NAMES/DATA sets without human intervention.

Each line in the "value file" (NAMES/DATA) can be 77 columns long. Each NAMES/DATA set can be 99 lines long. Imagine the possibilities!

Save this file under any filename: (i.e DSK1.NAMES/DATA)

3. LOAD FORMATTER

You now have a FORM LETTER file (DSK1.FORM-LETR) and a VALUE file (DSK1.NAMES/DATA).

Cycle Printer OFF/ON to cancel old printer codes

Load the FORMATTER and TYPE in answers to the questions and hit ENTER

(FILENAME:) DSK1.FORM-LETR
(PRINT DEVICE) PID.LF
(USE MAILING LIST?) Y
(WHICH LETTERS?) A (all or letters # 1,2,3,etc.)
(PAUSE AT END OF PAGE?) N (assumes tractor feed, for single sheets- Y)
(MAIL LIST NAME) DSK1.NAMES/DATA

4. Now sit back and wait for a FORM LETTER to be printed for each SET OF NAMES/DATA in your VALUE FILE. See Example #3 for MAIL READY letters. Have fun sending all your friends and relatives "customized" FORM LETTERS!!

5. Now study the TI-WRITER manual and learn how to produce other FORM LETTERS. You can use .ML(Mail List Value File Variables), .DP(Define Prompts Variables) and Non-Defined Prompts (unasigned Value File variables) in the same letter. More on this later.

6. DO IT NOW! Fire up your 99/4, copy Examples #1 and #2 and learn how to produce FORM LETTERS!!

EXAMPLE #1 -FORM LETTER

```
0001
0002
0003   .LM4;RM76;FI
0004
0005
0006   .IN+54
0007   August 8,1989
0008   .IN +0
0009   Dear *1*
0010
0011   Our records show you are &*5* behind in your JUG dues. If this is an
0012   error, please let us know.
0013   JUG dues are now only &$12 a year. For a dollar a month you can keep
0014   current with the major happenings in the TI99/4 world via newsletter,
0015   monthly meetings and unlimited "help" phone calls.
0016   Send &*6* to Phil (address below) and you will be paid up until &*7*.
0017
0018   Sincerely
0019
0020
0021   ------------------------------FOLD---------------------------------
0022
0023
0024   Phil Van Nordstrand^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^-----
0025   JUG Treasurer
0026   430 Shadow Dr.^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^stamp
0027   Seabrook TX 77586
0028   ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^-----
0029
0030
0031
0032
0033                                          EXAMPLE #2 -VALUE FILE
0034                                          ---------------------
0035   .IN +32                                1 John
0036   *1* *2*                                2 Doe
0037   *3*                                    3 4546 Coryell ST.
0038   *4*                                    4 League City TX, 77573
0039                                          5 3 months
0040                                          6 $12
0041                                          7 May, 1990
0042                                          *
0043                                          1 Jim
0044                                          2 Smith
0045   .IN +0                                 3 788 Diana
0046   ------------------------FOLD------- 4 Houston TX 77572
                                             5 4 months
                                             6 $12
                                             7 April, 1990
                                             *
                                             1 Jane
                                             2 Jones
                                             3 888 Pt Lookout Dr.
                                             4 Nassau Bay, TX 77058
                                             5 2 months
                                             6 $12
                                             7 June 1990
                                             *
```

CRYPTO SUMS

By Charles Martinez (ChasM)

It never ceases to amaze me that, ten years after its introduction, the TI99 is still living up to its name as a HOME computer. Offered to the public at a time when other small computers were scaled down business computers, build-it-yourself hobbyist computers (which were nearly impossible for the average layman to figure out), or were just plain toys, the TI has long since earned its niche in homes all over the world. Like a family pet, (we call our computer "Tippy") it has been a friend we and our children grew with and grew up with and which itself has grown up as well.

Perhaps the main reason that this is so is because the TI is so friendly. It adapts to a user of any level of computer literacy. When I first purchased my TI, among my first pieces of software were the "Number Magic" and "Video Games" cartridges, followed by the Basic tutorial cassette. My children learned how to count and add and I was introduced to the mysteries of programming. It was then that I began to realize just how powerful a learning tool the TI really was. Since then, there has been hardly a session spent sitting behind the console that I have not learned something new. "Tippy" may be an old dog, but she's still teaching me new tricks.

Today, the days of typing in programs from 99er Magazine, debugging and saving to audio tape seen light years away. My "system" is "fully configured," and my software library allows me to do everything from word processing to finding my favorite recipe for Turkey Tetrazzini. I'll share a little secret with you, though; I still get the urge to start with a blank screen with "ready" in the lower left corner and create something I've never seen before. Oh, it does't have to be complex or elegant, just different. This is my way of satisfying my need for adventure and, at the same time, practicing a kind of computer "Zen," where programmer and computer become one.

Having waxed poetic, I suppose I should let you in on why I'm writing what is my first article for my club's newsletter. It has always been my belief that programs written for the TI computer should be fun as well as useful. After all isn't a HOME computer a FUN computer? With this in mind, my #programming efforts have always been directed to the fun person lurking in all of us. Also, since I detest the agony of keyboarding in and debugging long programs, I always try to keep it simple. The bells and whistles can always be added later.

The accompanying program satisfies all the above criteria. It's short, (six sectors) runs in console basic or extended and will load quickly, even from cassette. Yes, I have a soft spot for those of you with "unexpanded systems." Not everybody was lucky enough to scarf up the PEB with drive and memory for $400 back in '83. But most importantly, it's fun.

The program randomizes an addition problem and then substitutes letters for the numbers, cryptogram style. Your mission, if you decide to accept it, is to decipher the problem in the fewest guesses. You don't have to worry about being caught or killed, and no secretary will disavow you. At the prompt, enter a letter followed by a comma, and the number you think the letter represents. If you are correct the letter will be replaced by the number. Keep trying until you have replaced all letters with their respective numbers, at which time the number of guesses used will be shown and you'll get the chance to Play Again. If you feel guilty having fun, you can pretend it's an educational program that teaches logical thinking.

This program is also a learning tool. You are invited to take it apart to see how it works. As it is a "bare bones" program, feel free to add your own graphics, sound or whatever bells and whistles that appeal to YOUR sense of adventure. (I have a version with speech, for example.)

And if you have one of your own short fun programs, I hope my primitive efforts have inspired you to share it with the rest of the TI Home Community. I'm sure there are a lot of closet programmers out there just itching to get published. So, hey-let's see if we can get a National Short Fun Program (NASFUP) movement going here. Till next time, "May the FUN be with you!"

```
90 REM CRYPTO SUMS BY
       CHARLES MARTINEZ

100 RANDOMIZE

110 GUESS=0

120 CA=0

130 CALL CLEAR

140 TOP=INT(RND*9000)+1000

150 BOT=INT(RND*9000)+1000

160 TOT=TOP+BOT

170 OPTION BASE 0

180 DIM L(9)

190 FOR X=0 TO 9

200 L(X)=INT(RND*26)+65

210 IF X=0 THEN 250

220 FOR TEST=0 TO X-1

230 IF L(X)=L(TEST)THEN 200

240 NEXT TEST

250 NEXT X

260 DIM NU(12)

270 DIM NUM$(12)

280 TOP$=STR$(TOP)

290 BOT$=STR$(BOT)

300 TOT$=STR$(TOT)

310 FOR X=1 TO 4

320 NUM$(X-1)=SEG$(TOP$,X,1)

330 NEXT X

340 FOR X=5 TO 8

350 NUM$(X-1)=SEG$(BOT$,X-4,
1)

360 NEXT X

370 LN=LEN(TOT$)

380 FOR X=9 TO 8+LN

390 NUM$(X-1)=SEG$(TOT$,X-8,
1)

400 NEXT X

410 FOR X=0 TO 7+LN

420 NU(X)=VAL(NUM$(X))

430 NEXT X

440 FOR V=0 TO 3

450 CALL HCHAR(10,10+V,L(NU(
V)))

460 NEXT V

470 CALL HCHAR(11,9,43)

480 FOR V=4 TO 7

490 CALL HCHAR(11,6+V,L(NU(V
)))

500 NEXT V

510 CALL HCHAR(12,9,45,5)

520 FOR V=8 TO 7+LN

530 IF LN=4 THEN 550

540 CA=1

550 CALL HCHAR(13,2+V-CA,L(N
U(V)))

560 NEXT V

570 FOR X=0 TO 7+LN

580 IF L(NU(X))>60 THEN 650

590 NEXT X

600 PRINT "GUESSES:";GUESS

610 PRINT : ;" DO YOU WANT T
O PLAY AGAIN?  (Y or N)"

620 INPUT ANS$

630 IF ANS$<>"N" THEN 110

640 END

650 PRINT "  ENTER YOUR GUES
S AS:         LETTER,NUMBER",

660 INPUT GL$,GN$

670 GN=VAL(GN$)

680 GL=ASC(GL$)

690 CALL CLEAR

700 GUESS=GUESS+1

710 IF GL=L(GN)THEN 730

720 GOTO 440

730 L(GN)=ASC(GN$)

740 GOTO 440
```

LI'L TUTOR

By Ed Machonis

LI'L TUTOR started life as a one liner, spawned a couple of sibling one liners, merged with them into a TIny Gram then, flushed with success, just kept on growing. Fortunately a publication deadline finally checked its growth. This is contrary to the way my programs usually grow.

This program is designed to help preschoolers recognize the letters of the alphabet and digits of our number system. Except for the initial menus, all keyboard entries are single key presses and an incorrect key press is ignored. Alpha lock can be in either position.

Only a cassette recorder and a speech synthesizer are required, although the program will run without speech, albeit very slowly. In this event all CALL SAY statements should be deleted to speed things up.

The initial menu allows a choice of letters or numbers. I know the program could have been written to teach both at the same time but I wouldn't relish the job of explaining to a child the difference between the number 0 and the letter O. I think it's best if letters are learned separately from numbers as different concepts are involved.

On the next menu, Option 1 provides for the sequential display of the characters selected as double sized stationary sprites. Tex pronounces the name of the character.

Option 2 is similar except that the characters are displayed in random order. A time delay is introduced before Tex speaks so that the tot has a chance to name the character before Tex does. As the child grows proficient, the delay can be reduced by changing the values for "T" in Line 30. Different values are used for letters and numbers because it takes Tex longer to think of the number names.

Option 3 lets the kid have at the keyboard. When a key is pressed the character is displayed on screen and named. Again a pause lets the kid beat Tex to the punch.

Option 4 randomly displays and names the character on screen. If the child presses the corresponding key, Tex speaks rewarding words and a new character is displayed and named. This option is also of value for After-shoolers who have trouble locating the letter keys.

```
10 ! ***** LI'L TUTOR *****
   *   By Ed Machonis   *
   * QB99ers Bayside NY *
   *********************

20 DISPLAY AT(10,1)ERASE ALL
:"1) LETTERS";;:"2) NUMBERS"
;;:"CHOICE?" :: ACCEPT AT(14
,9)VALIDATE("12")SIZE(1)BEEP
:C :: RANDOMIZE

30 IF C=1 THEN L=65 :: H=90
:: R=26 :: T=500 :: M$="A TO
 Z" :: N$="LETTERS" ELSE L=4
8 :: H=57 :: R=10 :: T=250 :
: M$="0 TO 9" :: N$="NUMBERS
"

40 DISPLAY AT(8,1)ERASE ALL:
"1) ";M$;;:"2) RANDOM ";N$;;
:"3) KEYBOARD ENTRY";;:"4) K
EYBOARD MATCH";;:"CHOICE?" :
: ACCEPT AT(16,9)VALIDATE("1
234")SIZE(1)BEEP:C

50 CALL CLEAR :: CALL SCREEN
(5):: ON C GOTO 60,70,90,110

60 FOR K=L TO H :: A$=CHR$(K
):: CALL MAGNIFY(2):: CALL S
PRITE(#1,K,16,85,120):: CALL
 SAY(A$):: FOR D=1 TO T :: N
EXT D :: NEXT K :: GOTO 60

70 J=INT(RND*R)+L :: IF J=K
THEN 70 ELSE K=J :: A$=CHR$(
K):: CALL MAGNIFY(2):: CALL
SPRITE(#1,K,16,85,120)

80 FOR D=1 TO T :: NEXT D ::
 CALL SAY(A$):: GOTO 70

90 CALL KEY(3,K,S):: IF K<L
OR K>H THEN 90 ELSE A$=CHR$(
K):: CALL MAGNIFY(2):: CALL
SPRITE(#1,K,16,85,120)

100 FOR D=1 TO T :: NEXT D :
: CALL SAY(A$):: GOTO 90

110 FOR I=1 TO 10 :: READ SP
$(I):: NEXT I

120 FOR D=1 TO 200 :: NEXT D
 :: J=INT(RND*R)+L :: IF J=K
 THEN 120 ELSE K=J :: A$=CHR
$(K):: CALL MAGNIFY(2):: CAL
L SPRITE(#1,K,16,85,120):: C
ALL SAY(A$)

130 CALL KEY(3,X,S):: IF S<1
 OR X<>K THEN 130 ELSE CALL
SAY(SP$(INT(RND*10)+1)):: GO
TO 120

140 DATA FINE,GOOD,VERY GOOD
,GOOD WORK,RIGHT,THAT IS COR
RECT,YOUR RIGHT,THAT IS RIGH
T,THAT IS EXACTLY RIGHT,YOUR
DOING FINE
```