CLEVELAND AREA TI-994/A USER GROUPS NEWSLETTER

NOVEMBER, 1991

OFFICERS	NORTHCOAST	TI-CHIPS	MEETING DATES	
CO-PRESIDENT	BOB KAGY 1-255-2609	MATT ANDEL 676-9759	NORTHCOAST 1:30 P.M.	TI-CHIPS 10 A.M.
CO-PRESIDENT		GLENN BERNASEK 238-6335	EUCLIDIAN ROOM	N.ROYALTON LIBRARY
TREASURER	FRANK JENKINS 283-8526	LIN SHAW 235-3912	EUCLID SQUARE MALL	STATE RD & RT 82
EMBERSHIP	MARTIN SMOLEY 1-257-1661	JOHN PARKEN 331-2830	THIRD SATURDAY	THIRD SATURDAY
	6149 BRYSON	4172 W. 217TH ST.	NOVEMBER 16, 1991	
	MENTOR, OH 44060	FAIRVIEW PARK, OH 44126	DECEMBER 21, 1991	
ECRETARY	BERNIE ZUCKERMAN 381-4088	DENNIS LIKENS 842-9627	JANUARY 21, 1992	
IBRARY(DISK)	MARTIN SMOLEY 1-257-1661	HARRY HOFFMAN 631-2354	FEBRUARY 15, 1992	
TAPE &MODS)	FRANK JENKINS 283-8526	JOHN PARKEN 331-2830	MARCH 21, 1992	
HARD COPY)	DICK ALDEN 1-352-9172		APRIL 18, 1992	

NOTE: As of Monday, October 28, I had not received Executive notes from either CHIPS or NorthCoast officers. I do know that Oscar Predmetsky gave a demo at CHIPS on using Hungarian characters in TI writer. I received the two pages enclosed too late for last month's newsletter. They are included this month so that you can see what unusual uses people make of their TI computer.

Wes Richardson gave a demo on writing XBASIC programs using variables. We only have room for part of this tutorial this month. Next month we will print a program using this tutorial.

I consider the following very important and am using it in place of the usual editorial comments.

CLEVELAND FREENET
Comments by Deanna Sheridan - Northcoast 99er

It has been a long time since I have reminded you that we have a TREASURE here in Cleveland that you are letting go to waste. As we have told you time and again, this service is indeed FREE, and the fame of the FREENET has started to spread throughout the country. Evidently you can connect to other systems and converse with people around the country...at least they can connect with US.

When I checked my mail box for Dennis Likens CHIPS notes this month (which weren't there), I went over to the TI SIG to see if anything was happening. Most of the time it is deader than a doornail.

There were several new postings, and I could hardly believe my eyes to find one from Barry Boone. I said to myself, this can't be THE Barry Boone. But as I read on, it was THE Barry Boone with the following message:

From: bd970@cleveland.Freenet.Edu (Barry E. Boone)
Subject: Hey 99'ers
Date: Tue Oct 22 12:27:31 1991

This is my first time up here, and I thought I'd just say ...I stumbled across this place while exploring Internet. I hafta admit that a TI SIG is the last thing I expected! Makes it really nice to be able to get up here via a terminal at school, so you will probably see me up here pretty often. I am getting ready to release numerous things for the 99/4A and 9640,including Sound F/X... a piece of software that will allow you to play digitized sound files on the 99/4A and Geneve (including files from the IBM, MAC, Amiga, etc.)... The software will allow you to use any and all memory you have, including 80column card RAM, Supercart, Superspace II (32K), Memex (for the 9640), etc. With a stock 99/4A, you get 34K of sound buffer, room enough for quite a bit. The quality of the sounds coming from your monitor is quite impressive... in many cases, it sounds just like a TV.

The package is available right now from me for $14.95 plus $2 for S&H. If you send $22 (includes shipping) I will send you Sound F/X and some disks full of sounds.

Barry Boone, Box 1233, Sand Springs, OK 74063

After Nov. 1, you may order Sound F/X through Texaments for $14.95 plus $3 S&H... Sound disks paks are $2.95 each (2 disks)

I will have the package on display at the Chicago Faire.
There will also be 10 video games released for the 9640!
Until next time,
Barry -- bd970
Internet (until Dec. 20, 1991):
U0309AK@vms.ucc.okstate.edu

It is a real shame that we have out-of-town people appreciate what we have here more than we do. I immediately sent Barry a reply, and had a reply the next day, which means

See back page

Why do we need foreign characters in America ?

I need it because I have,a lot of relatives in Europe and they do not speak English and when I write to them I have to write in Hungarian.

You need it my friend because EVERYTHING you buy is made in a foreign country (the clothes you wear, the pen you write with, your computer, all electronics and cameras, etc.etc.).

We are at the point when you buy some thing the instruction booklet that comes with it is printed out in three languages everything but English.

Recently my friend bought a telephoto lens for his camera; the lens was made in China and the instruction booklet was printed in three languages (Chinese, German and Spanish).

How does the TI-writer work with foreign characters what you can actually see on the screen as you type ?,.......

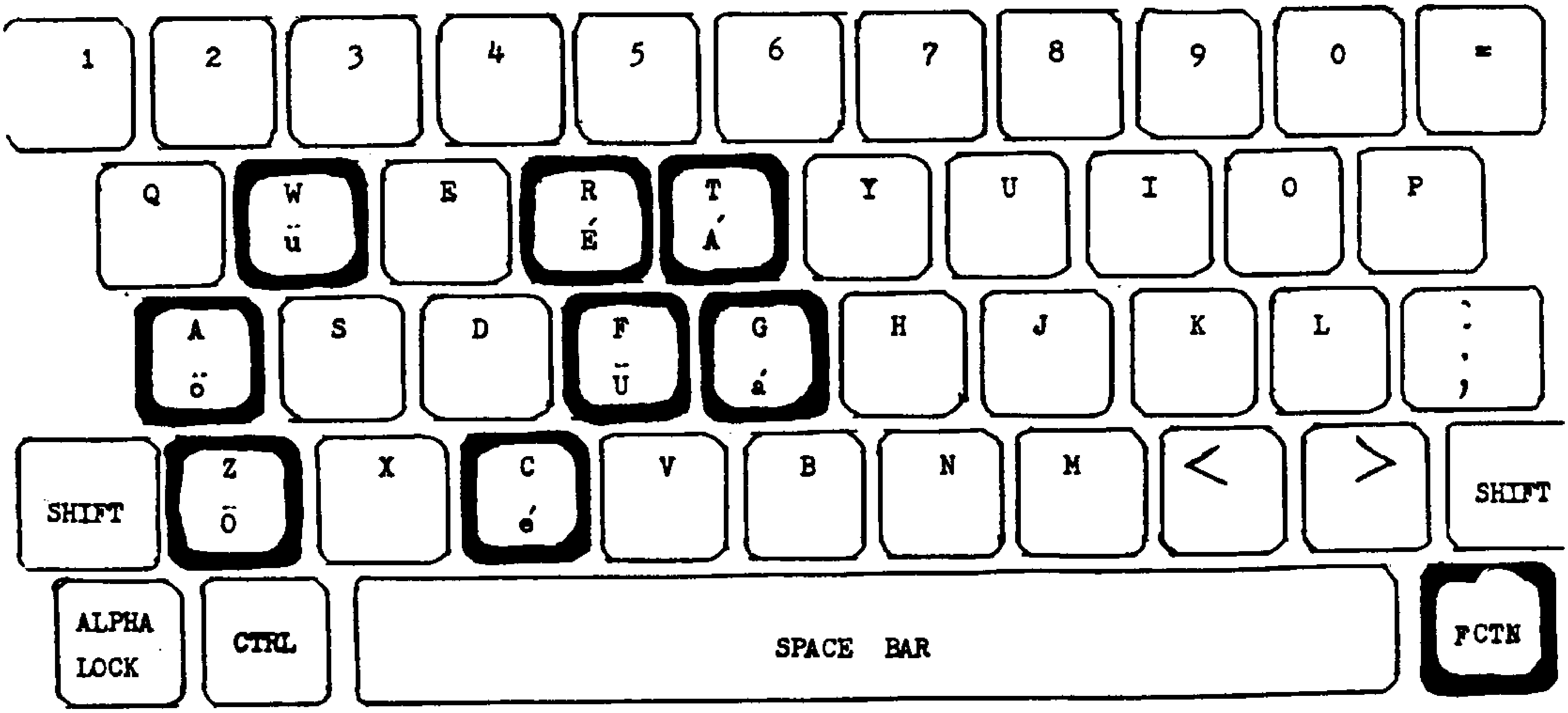
English; A a E e O o U u

Hungarian; A a Á á E e É é O o Ö ö U u Ü ü

I will explain everything at the meeting !

Oscar

The TI-99/4A keyboard



HUNGARIAN GULYAS

This is a famous Hungarian gulas receipe which is about 1000 years old unchanged and it takes about 4-hours to cook on a slow heat. If you try to speed it up it will turn into a hamburger !

At this comeing TI-chips meeting I will demonstrate that the TI 994A can handle a very difficult Hungarian language (which requires SEVERAL foreign language character sets together). I will make the TI print this receipe to the screen and to the printer in Hungarian language on the same page mixed with English, and because I know that not to many people can read Hungarian,....... the only way they will understand it completly without any doubt is if I bring a pot full of Hungarian gulas to the meeting (which I will do !).

Here is the recipe for it,...... you need;.......

0.5 kg marha hus,........... 2-szál zeller,.............. 1- kg zsir
1-db friss paradicsom,......... 1- nagy hagyma,........ 2-db krumpli
1-db zöld paprika,.............. 1/4-nagykanál piros paprika por
2-db sárgarépa,....... 1-nagykanál so,....... 2-db fehér répa és a zöldjei.

Elöször a felapritott hagymát kel megpuhittani zsirban, akkor hozzá adni a kis kockákra felvágott hust, a felapritott paradicsomot, sot, paprikát és 1/4-pohár vizet és ezt lassu tüzön fözni kell 3-ora hosszat (amig a hus teljessen meg nem puhul).

Akkor hozzá kell adni minden felszeletelt zöldséget (csak a krumplit nem) és 1-pohár vizet és megint lassan fözni 1/2-ora hosszat.

Utoljára hozzá kell adni a kockákra felapritott krumplit és frissen csinált nokedlit és 1-liter vizet és megint az egésszet fözni kell 10-percig (vagy amig a krumpli meg nem puhul).

That is it folks,...... and now let me guess; you want it in English,...... but if I give it to you in English then it would be to simple and even an IBM computer would be able to run it and turn my slow cooking gulas into a high speed hamburger.

I know, I know, now adays all computers can run a foreign character set such as a; German or Spanish or Italian, Etc., but can they write Hungarian ?, can they mix SEVERAL foreign character sets and English ON THE SCREEN and PRINT THEM OUT on paper at the same time (without switching in and out of sub programs, without switching dip switches on the printer) like this; A a Á á E e É é O o Ö ö U u Ü ü ?,...... the TI 994A can !,.. how ?,... by useing a modified CHARA1 file (which works with the TI-writer automatically),... whos got it ?,.... I do !

Oscar Predmetsky
(TI-chips)

1991 october

TI-Base -By- Inscebot, Inc.
P.O.Box 291610, Pt.Orange, FL 32129
Version 3.01 Tutorial 24.1.1 By *Martin A. Smoley*

NorthCoast 99`ers User Group - Aug. 17, 1991

Printer Controls

There is a printer Database (Db) located on the TI-Base (TIB) program disk. This Db contains X type fields which are tricky to work with, but are quite useful if mastered. I have listed my PRINTER Db at the bottom of this page for your inspection. Note: I have chopped off the length of the BLANK field so it would fit on the page. The name of this Db is simply PRINTER and is invoked by a line in the SETUP CF that consists of the word PRINTER followed by the name of your printer. In my case it's PRINTER EPSON. This is the line you will see in the SETUP CF that comes from Inscebot, but there will be a * (asterisk) at the beginning of the line. The asterisk tells TIB that this is a comment line so it is not executed. If you remove the asterisk from that line the PRINTER Db will be used by TIB, the data in the EPSON (or whatever) section will be moved to active memory and the PRINTER Db will be closed. At that point (if your printer is ready) you can send printer commands with a simple (FF) for Form Feed, or (CR) for Carriage Return, etc.

Marty's PRINTER Database info

CREATED CHANGED 04/30/89

FIELD	DESCRIPTOR	TYPE	WIDTH	DEC
1	NAME	C	010	
2	FF	X	002	
3	LF	X	002	
4	CR	X	002	
5	G	X	004	
6	UL	X	006	
7	e	X	002	
8	f	X	004	
9	4	X	004	
10	E	X	004	
11	SPS	X	006	
12	SBS	X	006	
13	HT	X	002	
14	ST	X	020	
15	Drft	X	026	
16	BLANK	X	030	

003 1 PRINTER 00006/00008

Remember, until you remove the asterisk (*) from the beginning of the "* PRINTER EPSON" line in the SETUP CF and rerun the SETUP CF or restart your system, none of this stuff will be available to you. "OK, the PRINTER Db." The PRINTER Db is just another Db. If you follow the rules you can use it to great advantage. [The Rules!] The first FIELD must be NAME, it must have a WIDTH of 10, TYPE must be "C" and the PRINTER Db must be SORTed ON NAME. All of the other FIELDs must be X type and must be even in length (WIDTH). NOTE: They can be 2,4,6,8,10,26,30 etc., they cannot be 3,9,11,29 etc. In the DESCRIPTOR column, NAME must be first, "FF" must be second, "LF" must be third and "CR" must be fourth. After that you can put whatever you want in the DESCRIPTOR column.

USE DSK1.PRINTER <E>
MODIFY STRUCTURE <E>
Make all the changes you need.
(Fctn 8) <E>
Select [USE DATA AS-IS]
Press ENTER

IMPORTANT! Never use (Fctn =) to Quit or turn off your system to leave TI-Base. The way to leave TIB is to type QUIT and press ENTER "<E>" at the Dot prompt (Dp). Quiting TIB by any other means could destroy an open Db.

The USE sequence above will help you modify your PRINTER structure (as I have). "If you want to leave the structure as it comes from Inscebot, that's OK too. The things I'm telling you are just suggestions, not commands." To modify the data in the PRINTER Db you can USE it and then type EDIT <E> instead of the Modify Structure. I will try to give you some idea as to my modifications of the PRINTER Db. On the STAR NX 1000 printer I use for data printing, (ESC E) starts Emphasized print mode. So I changed that DISCRIPTOR to simply "E". You can see it as item 10 to the left. Then I EDITed the data in my PRINTER Db so the RECord for EPSON under E contained 1B45. This is the actual data that will be sent to the printer when I issue a (E). So what is 1B45? If you EDIT the data and type in 1B45, you enter it exactly as it looks, (one, "B", forty-five). However, TIB sees it as two bytes of Hex. The first byte is 1B, which is Hex for 27, which means <ESC>. The second byte is 45, which is Hex for 69, which is decimal for "E", which is where we started. If you look in your NX 1000 printer manual, around page 61, you'll see Emphasized printing [ASCII = <ESC> "E"], [Decimal = 27 69], [Hexidecimal = 1B 45]. You will find this information in almost all printer manuals. Enter the stuff under Hexidecimal as data in the PRINTER Db (using NO SPACES).

Next Page. ➜

```
REC  NAME        FF LF CR G    UL     e  f    4    E    SPS    SBS    HT ST                   Drft                       BLANK
0006 DIABO       0C 0A 0D 0000 000000 00 0000 0000 0000 000000 000000 09 00000000000000000000 00000000000000000000000000 00000000000
0000 EPSON       0C 0A 0D 1B47 1B2D01 0E 000F 1B34 1B45 1B5300 1B5301 09 1B440000000000000000 1B481B2D0014121B351B461B54 00000000000
---- ---------- -- -- -- ---- ------ -- ---- ---- ---- ------ ------ -- -------------------- ---|-----|-|-|---|---|---| -----------
0003 MX-80       0C 0A 0D 1B47 000000 0E 000F 0000 1B45 000000 000000 09 1B440A0A0A0A0A0A0A00 1B4814121B4600000000000000 00000000000
0002 NEC         0C 0A 0D 0000 000000 0E 1B51 0000 1B21 000000 000000 09 1B283031302C30320000 1B220F1B4E0000000000000000 00000000000
0004 OKIDATA     0C 0A 0D 1B48 1B4300 1F 001D 0000 1B54 001B4A 001B4C 09 00000000000000000000 00000000000000000000000000 00000000000
0005 PROPRINTER  0C 0A 0D 0000 000000 00 0000 0000 0000 000000 000000 09 00000000000000000000 00000000000000000000000000 00000000000
0007 SEIKOSHA    0C 0A 0D 1B23 1B5800 0E 1B43 1B42 1B48 1B5500 1B6800 00 00000000000000000000 1B241B590F1B4E000000000000 00000000000
0001 TI-850      0C 0A 0D 1B47 000000 0E 000F 0000 1B45 000000 000000 09 1B440A0A0A0A0A0A0A00 1B4814121B4600000000000000 00010000000
```

TI-Base - By - **Inscebot, Inc.**
P.O.Box 291610, Pt.Orange, FL 32129

Version 3.01 Tutorial 24.1.2 By ***Martin A. Smoley***

NorthCoast 99`ers User Group - Sept 14, 1991

In my PRINTER Db (G) stands for Double-strike, (UL) stands for Start underlining, (e) stands for Expanded print one line, (f) stands for Condensed print, (4) = Italic, (E) = Emphasized, (SPS) = Superscript, (SBS) = Subscript, (HT) = Horizontal tab, (ST) = Set horizontal tab stops every n columns and (Drft) stands for return to Draft Mode, or turn off everything I described previously. Drft is the hard one of the bunch. If you make any changes to the ON commands I just mentioned, you will also need to change the commands that turn off that printer change. The OFF commands are all lumped together in Drft. I have put a dashed line under the EPSON section and I have placed vertical marks under the last character of the individual commands in the Drft field. You can compare these commands to the OFF commands in the manual that came with your printer. If you want to produce nice looking reports, with columns of numbers or whatever, you will need to completely understand your printer handbook commands and how to install them in this portion of TIBs command system. I have listed one of my many label programs on this page to give you some idea of the use of the printer commands within TIB. All the print stuff is done near the end of the CF. This CF may use Condensed, normal sized Emphasized and possibly Expanded Emphasized print on a single label, depending on the overall length of the persons name which it finds in Name Length [NL]. This CF must be typed in using FunnelWeb or TI-Writer as it is too big for TIB's MODIFY COMMAND setup. In order to use it you need a printout of the Db to be able to enter the persons REC number when asked. "INTERESTING NOTE", [See GO RCNM], This command i.e. GO 4, GO 78 will locate a record from a CF or from the Dp. It will GO to that RECord no matter how the Db is sorted.

***1LBL91/C**ommand File*

```
   *          09/01/91      1LBL91/C
CLOSE ALL
SET TALK OFF
SET HEADING OFF
SET RECNUM OFF
SET PAGE=000
SET PRINTER=RS232.CR.LF.DA=8
CLEAR
WRITE 10,4,"RS232.CR.LF.DA=8"
WRITE 12,4,"Turn the printer on now!"
WAIT 3
WRITE 10,4,"                "
  ITE 12,4,"                          "
   USE DSK2.CLUB91
LOCAL TEMP1 C 60
LOCAL TEMP2 C 39
LOCAL TEMP3 C 40
LOCAL ANS N 2 0
LOCAL RCNM N 4 0
 WRITE 2,4,"To quit this Command File"
 WRITE 4,4,"Enter Record Number:  999"
WHILE .NOT. (EOF)
 WRITE 10,4,"Enter the Record Number: "
 READ 10,30,RCNM
 WRITE 2,4,"                           "
 WRITE 4,4,"                           "
 GO RCNM
  IF (EOF)
   CLOSE ALL
   RETURN
  ENDIF
   REPLACE TEMP2 WITH TRIM(FN) ¦ " ";
           ¦ MI ¦ " " ¦ LN
 WRITE 10,4,TEMP2
 WRITE 14,4,"0 = no labels"
 WRITE 16,4,"How namy Labels:"
 READ 16,21,ANS
 WRITE 10,4,"        Printing Labels   "
*
   REPLACE TEMP1 WITH "                  ";
¦ "                     Exp. Date: " ¦ XP
 IF MI = "  "
   REPLACE TEMP2 WITH TRIM(FN) ¦ " ";
                        ¦ LN
  ELSE
   REPLACE TEMP2 WITH TRIM(FN) ¦ " ";
           ¦ MI ¦ " " ¦ LN
  ENDIF
   REPLACE TEMP3 WITH TRIM(CT) ¦ " ";
           ¦ ST ¦ "    " ¦ ZP
  WHILE ANS > 0
     PRINT (Drft),(f),(G),TEMP1,(LF)
    IF NL < 17
     PRINT (Drft),(E),(G),(e),TEMP2
     ELSE
     PRINT (Drft),(E),(G),TEMP2
     ENDIF
     PRINT SA
     PRINT TEMP3,(LF)
   REPLACE ANS WITH ANS - 1
  ENDWHILE
ENDWHILE
CLEAR
CLOSE ALL
SET PRINTER=PIO.CR.LF
SET TALK ON
SET RECNUM ON
SET HEADING ON
RETURN Copyright Martin A. Smoley 1991
*
* 1LBL91  Save as 1LBL91/C  09/01/91
* ******       Prints all Labels
*          USEs CLUB91 ----- DSK2.
* Prints to TI/Epson Printer - RS232
```

Good luck with TI-Base.   Marty.

Maybe Next Month

WRITING PROGRAMS PART 02 VARIABLES

by WESLEY R. RICHARDSON
NORTHCOAST 99ER'S, CLEVELAND, OH

PURPOSE

The intent of this article is to help programmers understand the use of constants and variables, including arrays, in TI Extended BASIC.

SCOPE

While many of the concepts regarding variables apply to any programming language, the specific commands, functions and statements in this article refer to those on the Texas Instruments (TI) computer, TI-99/4A, using the Extended BASIC (XB) language.

INTRODUCTION

Constants and variables are parameters. The advantage of using parameters is they allow a program to be written which will work even if the person writing the program does not know which numbers or words you will use in the program when you run it. Examples of this concept are the handheld calculator which uses numeric variables that you entered when you multiplied two numbers, and a word processor which will work with any words you type.

After the constants and variables are entered in a program, the parameters are used by an expression to create an output, or to make a decision based on your input. Since most computer programs do not modify themselves while running, the use of variables in the program gives it the flexibility and power to be used in many different applications.

STRING CONSTANTS

A constant is a parameter whose value cannot change during the running of the program. A string constant is a group of alphanumeric characters up to 153 characters long. They are usually enclosed in quotation marks, but are not required to be in quotes. String constants have two primary uses in programming, one in relational expressions to evaluate a string variable, see a), and the other in forming new strings for displays or printing, see b).

```
a) IF A$="Y" THEN 300
b) B$=C$&" PAGE NUMBER IS "
```

The letters and spaces between the quotation marks comprise the two string constants in these examples. I am already getting ahead of myself, because I have used string expressions which I will discuss later.

NUMERIC CONSTANTS

Numeric constants are similar to string constants, but numeric means they are numbers. Since they are constants, again they cannot change during the program, because if they did, we would call them variables instead of constants. How can something that is constant be useful in a program? Here are two examples of how constants could be used:

```
d) IF A=4 THEN 300
e) B=5*C
```

The numeric variable A is compared with the numeric constant 4 to make a decision. The numeric variable C is multiplied by the constant 5 and B is set equal to the result. I guess I just can't write about one part of parameters without introducing the next.

STRING VARIABLES

Each programming language has different rules which define allowable variable types. Extended BASIC on the TI has one of the most flexible requirements of all computers. Variables can have names of up to 15 characters long, with the first character any of the letters of the alphabet, the "at" (@) symbol or the "underline" (_) symbol. After the first character, the remaining characters may also include any of the numbers. For string variables, the last character must be a dollar ($) sign. This limits the number of string variable names to about 10,000,000,000,000,000,000,000 while numeric variables can be selected from a number a little larger than 38 times that number. Examples of string variable names are:

```
f) A1234567$ and _A1B2C3D4E5F6G$
```

Note that the TI converts all lowercase letters to uppercase when they are used in a variable name.

One further classification of variables is whether they are scalar or array type of variables. Array variables are discussed later. A scalar variable means a "real" number in mathematics. On the TI-99/4A, a scalar string variable is a string variable whose value is any single string of alphanumeric characters up to 255 characters long. Examples of scalar string values are:

g) "1234567" and ABCDEF123456

NUMERIC VARIABLES

The names of numeric variables in Extended BASIC have the same specifications as do the string variable names, except the dollar sign ($) is not used. Thus, a numeric variable name can be up to 15 letters or numbers long, but cannot start with a digit. Examples of numeric variable names are:

h) A1234567 and _A1B2C3D4E5F6G7

The values which a numeric variable may be are determined by the numeric range of the computer. In Extended BASIC, the largest number which can be printed is 9.99999999E127 and the smallest is 1.00000000E-128. Normally XB will only print or display numbers up to E99, with a maximum of 10 characters, however if you use an IMAGE statement or a PRINT USING or DISPLAY USING, with 5 circumflex symbols (^) for the exponential term, numbers with exponents of +-100 or more can be displayed or printed. Examples of numeric variable values are:

i) 9.345 and -123.456 and 3.45678E-37

STRING ARRAYS

Arrays are a method of notation which allows a programmer to refer to or use a group of variable names and values in such a way that it greatly simplifies programming. There are many analogies of arrays which we use in our everyday life, and I would like to present one of those, which will perhaps help you to understand arrays.

If I wished to direct someone to a piece of information which was on a certain page of a document, I could say look on page 4, for example. The variable name is the name of the document, so the reader knows which document I am referring to. I might also say to read pages 4 through 7 of that document. As long as I gave the document name and the page numbers, the reader should be able to read the pages I indicated. The document is then like a one dimensional array, with each page being a different, but ordered element of that array.

Now what if I had several documents, all of which related to a particular subject. I could give them each a different name or variable name, or I could assign each a sequential number, and give them a common name such as TAXFILE. If I then told you to read TAXFILE, document 1991, page 3, you would know exactly where to look. This is like a two dimensional array. The year sequence is one dimension, and the page is the second dimension. If all of my tax records are in drawer one of a file cabinet, and all of your records are stored in drawer two, and other records are in drawers three and four, then I have created a three dimensional array of tax information. I can uniquely identify which page I refer to by giving drawer number, year, and page number. Of course this can be extended to as many dimensions as we wish to use.

To use an array in Extended BASIC, you first define it with a DIM statement such as DIM TAX$(23,5,4). The name for an array must follow the same conventions as do any other string or numeric variables. In the DIM statement you also tell the maximum number of elements for each of the dimensions. This is slightly modified by the OPTION BASE statement which defines whether you start counting elements at zero or one. The default is 0 if you do not use OPTION BASE. When you use the array, you tell the program which element you want to use by the values within the parentheses. TAX$(2,3,4) is thus referring to the string value contained in the second drawer, the third document, and the fourth page.

Arrays in XB can be up to seven dimensions. The most common in programming is one dimension, followed by two, three and higher dimensions. The other constraint is the memory space available. For the TI, the memory limits arrays to about 150 string elements for most programs, but this greatly depends upon how many characters are in each element. Large computers can be programmed with arrays of hundreds and even thousands of dimensions. This many dimensions is useful for demographic analysis. A way to determine how much memory you are using is to run your program, fill the array with typical strings, and then do a FCTN4 break, then type SIZE to see how much stack and program memory is free. When your program gets under 1,000 bytes of stack free, you will need to be careful about adding any more variables or increasing array sizes.

An element of a string array contains a string value which can be used in exactly the same way that a scalar string variable returns a string value.

NUMERIC ARRAYS

Numeric arrays use the name conventions of numeric variables, with the additional numbers determining which element we are referring to. As stated under string arrays, all arrays must be first defined with a DIM statement before they can be used. Numeric arrays evaluate or return a number whose value was stored in the element or position indicated.

Now back to the advantages of arrays for the programmer. I can write a loop to check each element to see if it has a certain value or to print all of the values to the screen. a loop like: FOR I=1 TO 10 :: PRINT WEIGHT(I) :: NEXT I provides a shorthand notation to refer to all 10 elements or values of weight. This same method could be used to see if your name appeared on any page of TAXFILE from above.

Two arrays can be used together to maintain data which is related. For example a string array may have last names and first names, and a parallel numeric array may contain wages, ages, and hours worked. When used in a program, the array index can point to an element in each array which contain information about the same subject.

STRING EXPRESSIONS

Now that we know what string constants and variables are, how can we use them? I will use A$, B$, and C$ to represent any parameter which evaluates to a string value for the following discussion.

Of course we can simply print or display the string A$. We can also combine strings using & for concatenation. If A$="HI " and B$="BOB", and C$=A$&B$, then C$="HI BOB". This provides a way to combine any strings which we select. We can repeat a string A$ using C$=RPT$(A$,3) and get C$="HI HI HI ". The opposite of combining strings is to parse them. The SEG$ function combined with the LEN and POS functions allow selection of any subset of letters from another string. LEN is useful to tell us how long the string is, and POS to locate a particular character, such as a space character. If A$="FRIDAY", then SEG$(A$,1,3) returns "FRI".

The ASC(A$) function will return the ASCII code of the first character of the string A$. We can also use the VAL(A$) to convert a string to a number, assuming the string contains the proper characters.

NUMERIC EXPRESSIONS

Numeric expressions operate on any numeric parameters to return another numeric parameter. The primary numeric operators are the plus (+), minus (-), multiply (*), divide (/), and exponentiation (^). In addition, the following functions operate on numeric expressions to return a numeric expression: ABS, ATN, COS, EXP, INT, LET, LOG, MAX, MIN, PI, RANDOMIZE, RND, SGN, SIN, SQR, and TAN. Both numeric variables and numeric expressions be used in loops, as array element pointers and in statements and functions which require numeric expressions as inputs.

The functions CHR$ and STR$ use numeric inputs and return string values.

RELATIONAL EXPRESSIONS

The relational expression operators are: = < > <> <= >= which stand for equal, less than, greater than, not equal, less than or equal, and greater than or equal. These operators may be used on both numeric expressions and string expressions, but not at the same time. The result will be -1 if the test is true and will be 0 if it is false. The output value of a relational expression can be used as a numeric value or as a branching test like an IF-THEN statement. Examples of the use of relational expressions would be to test a string to see if it matched the string you were looking for in a list or array. You could use a relational expression to speed up a game if the player's score exceeded a certain value.

LOGICAL EXPRESSIONS

The logical expression operators include NOT, XOR, AND, and OR, in decreasing order of precedence. This means that in the absence of parentheses, perform the NOT operator first, then XOR, then AND, then OR when evaluating a logical expression. Logical expressions form the basis for Boolean algebra. Logical expressions, like relational expressions are used to evaluate relationships and when used that way, return -1 if the test is true and 0 if false. Logical expressions can also be used on the integer numbers in the range of -32,768 to 32,767 and when used in this way, return the decimal result of performing the operator on the binary value of the initial number(s).

CONCLUSION

Parameters, including constants, scalar variables and array variables allow programs the flexibility and power to utilize numbers and strings in a variety of ways. These include decision making for program branching, operating on groups of similar information, and performing calculations with numeric data. Variables are the means for all input and output from a program.

911018_WR_PROGRAM02

CHARLES W. GOOD
Box 647
Venedocia OH 45894

419-667-3131

October 17, 1991

Deanna Sheridan
Editor: CLEVELAND AREA 99/4A USER GROUPS NEWSLETTER
20311 Lake Road
Rocky River OH 44116

LETTER TO THE EDITOR:
Dear Deanna,

I was somewhat diamayed to read the comments of Les Kee in the minutes of the 21 Sept. 1991 TI CHIPS meeting, as published in your October 1991 newsletter. "Les Kee recommended that a letter be sent to the authors of FUNNELWEB that the program is mature and needs no further programming cause the more versions that come out, the more utilities are dropped."

If such a letter is written, THAT IS EXACTLY WHAT WILL HAPPEN. Tony McGovern's philosophy concerning improvements and bug fixes to Funnelweb are spelled out in his letter published in the Sept. 1991 issue of the Lima UG's newsletter. Tony has been contemplating for some time purchasing an Amiga and programming for it instead of the 99/4A. One of the reasons he hasn't yet is financial. He recently had a second corneal transplant in one of his eyes. This is expensive. In a recent letter to me, Tony McGovern says, "I sure am tired of hearing 'Oh No! not another revision of Funnelweb.' My only answer to that sort of comment is 'OK, I'll stop correcting bugs and adding new features.' And I mean it. The threat is real."

I know of nothing significant that was in Funnelweb v4.31 and is not in the current v4.40. For 40 column users there are two significant new features added to v4.40. DSKU file notes can be copied using v4.40's 40 column DISK REVIEW. Also, using file ML, it is possible to boot Funnelweb from the EA or MM modules, or as a CALL from a ramdisk and have an immediate screen display of 24 programs from which to to select. Just use the arrow keys to move the cursor to the item of your choice and press <enter> to run the program. This new feature is potentially very significant and labor saving. You can include everything on both of Funnelweb's central menus plus many additional items on this 24 item menu.

Unlike commercial software, you don't have to pay for new Funnelweb updates. Fairware is "try before you buy." It is not necessary for Funnelweb users to pay for each new update unless the user feels that the new features in the update are worth paying for.

I sure hope Les Kee's letter proposed letter is never written. We don't want to lose one of the most important programmers left in the TI community.

Charley

Charles Good
Newsletter Editor, Lima OH User Group

that he will be here a lot.

In case any of you are still asking 'Who is Barry Boone?', he has done lots of freeware stuff, including ARCHIVES which almost all of us use, but how many have paid for it. He did GIF99 as freeware and then made it commercial with Texaments. And you can see he is still developing packages. Just like Peter Hoddie, Paul Charlton and Dennis Faherty, he is a college student who has done wonderous things for the TI.

We have a great newsletter here in Cleveland not because we have a great newsletter editor, but because we have participation by members caring enough to write articles for that newsletter. We could have a great SIG on the freenet if we would have a few people drop by and leave a line. If you have a programming or hardware problem, drop a line and see if someone will answer it. It is very discouraging to go in find only 'For SALE' items. Tell something interesting that happened at the last meeting. This might entice people who are not members of either group, but are on the FREENET to join. I converse all the time with people who have TI's whose names I do not recognize. I tried for a long time to keep up with news over there, but got tired of being the ONLY person to care enough to put in my two cents worth. Also, between working on the newsletter and other projects, I end up with something called lack of time.

Following is another message that might interest you. I don't do enough on FREENET to even know what USENET is, but it sounds like a great way to scrounge up some interest.

Newsgroups: freenet.sci.comp.ti.general
From: rfd@po.CWRU.Edu (Richard F. Drushel)
Subject: Re: To the Sysops of this board
Date: Fri Sep 13 10:16:06 1991

Well, you might do what I did, when the Sysops of the ADAM forum were too busy with their own ADAM BBSes to call in here regularly--I became a Co-Sysop. I put a few feeler posts out on USENET about the ADAM (in comp.misc, comp.os.cpm and comp.os.misc) and in a week I heard from long-lost ADAMites all over the country.

You 99ers are lucky, there is a comp.sys.ti newsgroup, or something like that...no such luck for the ADAM.

I regularly check in on the 8-bit computer SIGs, just because they *ARE* likely targets for deletion due to neglect. Apple II, Commodore and Atari seem to be doing okay, I am trying to revive ADAM (I have enticed one new user), and Timex/Sinclair and TI99 seem...too quiet.

I posted a request on the Sysops SIG board #3, asking all Sysops to consider enabling the login/readmail times for their xx accounts, so that users could see when was the last time a Sysop maintained the account. So far, no response there, either.

Regardless of system, we remaining 8-bitters need to stick together, and generate some public *ENTHUSIASM* for our machines. Even if most of the posts are from the Sysops themselves, the information is bound to bring some of the many lurkers to the surface.

This is an important discussion thread, I think. I would be glad to pursue it with any interested parties in any public forum or in E-mail.

Regards, Rich Drushel ** CWRU Biology Ph.D. Student ** CFN

It seems that OTHERS care more than we do about our SIG on the FREENET. Common guys and gals(?). We have a real coup here with Barry Bonne. Get with it!

CLEVELAND AREA 99/4A USERS GROUPS
C/0 DEANNA SHERIDAN
20311 LAKE ROAD
ROCKY RIVER, OH 44116

CHECK YOUR EXPIRATION DATE.
THIS MAY BE YOUR LAST ISSUE!

FIRST CLASS

FIRST CLASS

FIRST CLASS