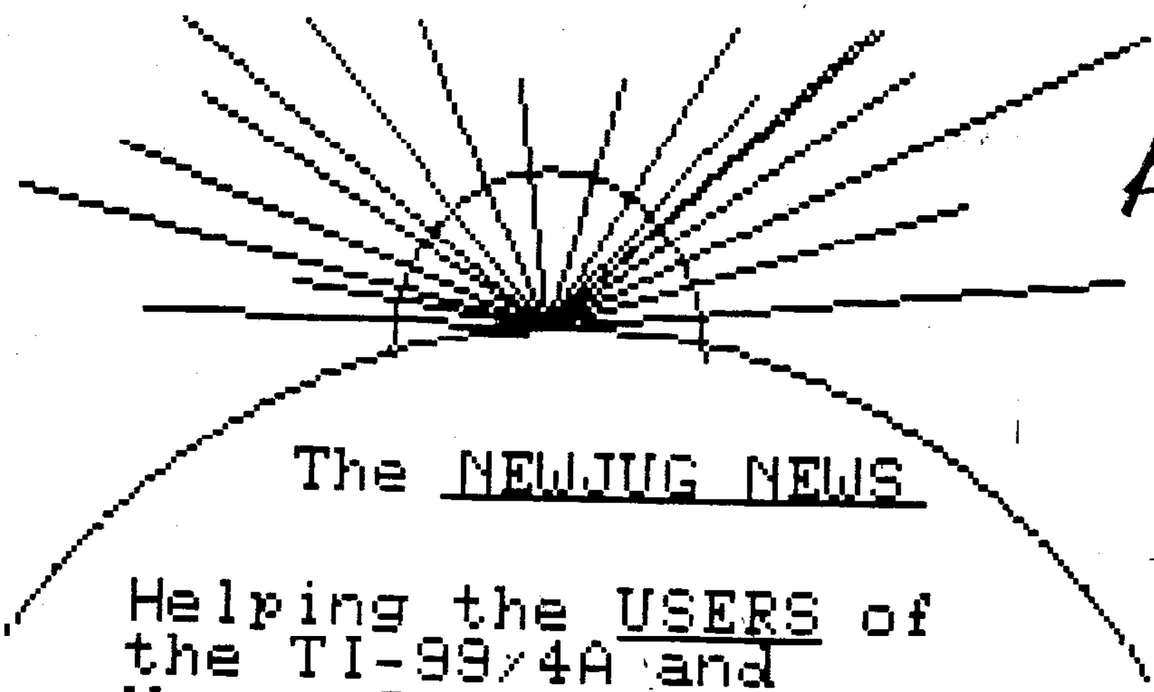

The NEWJUG NEWS

Helping the USERS of the TI-99/4A and Myarc Geneve 9640 into the dawn of a new decade.

NEXT MEETING:

MARCH 1, 1988

NEWJUG NEWS
c/o Bret Musser
60 Broadway Road
Warren, NJ 07060

NEWSLETTER EXCHANGES: PLEASE NOTE NEW ADDRESS!

Dallas TI Home Computer Group 9/9
P.O. BOX 29863
Dallas, TX 75229

NEWJUG MEETING SCHEDULE

======================

1988 The main meeting will start at 7:30 in the Iselin Public Library on Green St. in Iselin. Come early to get settled down for the meeting and for an informal discussion period on any topic of your choice.

FEBRUARY 2
MARCH 1
APRIL 5
MAY 3
JUNE 7

NEWJUG Club Officers

President:	Herb Oppenheimer..992-9734
Vice President:	Dave Green........463-9133
	Dan Gazsy.........???-????
	John Molinelli....???-????
	Bret Musser.......647-1437
Rec.Secretary:	Johan Nykvist.....727-6217
Corr.Secretary:	Stan Rosenthal....647-2365
Treasurer:	Mac Rochan........463-1918
Software Librarian:	Dave Green........463-9133
Newsletter Editor:	Bret Musser.......647-1437

* * * * * * * * * * * * * * * * * *

Please send all mail to:

NEWJUG NEWS
c/o Bret Musser
60 Broadway Rd.
Warren, NJ 07060

or

NEWJUG Corresponding Sec.
Stan Rosenthal
40 Cottage Place
Gilette, NJ 07933

Membership dues: $15 for individual
$20 for family membership

Send all dues to: Mac Rochan
6 Sunburst Lane
Piscataway, NJ 08854

If possible, pay your dues at the meeting.

****

The mentioning of a product in the NEWJUG News does not constitute an endorsement by the NEWJUG News or by NEWJUG itself. Opinions expressed here belong solely to the author of such opinions and do not represent the opinions of NEWJUG or the NEWJUG News. Copies may be made of this newsletter as long as the original author(s) is recognized.

Before I introduce this month, I must notify you all of a recent happening: earlier in January, Ed Zelazny, a good member and friend, died from a heart attack. Many of us will miss him.

Introduction: This is February.

Well, here we are into a new month. For me, January was one of my most trying months, with Mid-year exams and all. But now it is over, and my life can resume, and so can my excellence in writing this newsletter, which has been lacking the past month, and is lacking also in this issue. There is a reason: I have not used my computer that much. For that matter, I haven't used ANY computer that much. Usually, I log on about 10 hours minimum per week, but this month, I averaged one hour per week!

To be honest, I wrote some of these articles (and this introduction) just two nights before the meeting, and had the newsletter printed the day before the meeting! Why? Well, it is some form of self-torture...but not entirely voluntary. First, I start to write in the morning, then get involved doing something else for someone else, and then the next thing I know, its 10 o'clock p.m.!

Anyways, enough about me, and more about what is going to happen in the coming months. Firstly, mainly, and most importantly (now there's an illegal use of adverbs if I ever saw one) we are going to be preparing for TICOFF. We should really run a demonstration of something that our club has done, and we must distribute ads for the club. I think that when I took over this newsletter, I mentioned that as what we should make our Number One goal, and I'm sure I mentioned it last month.

Speaking of last month, I have some notes about the January newsletter. First, in the Geneve column, the parts relating to someone else's reactions to the Geneve were inappropriately written. I was not writing it for you, but for a few selected people who would probably never read this newsletter. I should have rewritten it once I regained my mind. Second, in the article about our future: all of what I said was logical, but I must emphasize one point--none of that has to come true. We CAN change that grim future I mentioned. (For a touch of reality in this case, change would most likely mean delay.) First, we must get all of the TI's out of the closet and back in operating condition. This also means getting the USER's of those machine back and running again. Second, participation in groups and for the newsletters of the TI world must increase. It has been for our group, and I am very thankful for that, but participaiton must increase universally. We cannot take and take from a system and not return anything. Look at our world today for proof of that. I just hope that no one drops the TI because of what I said. Thankyou Stan.

Anyways, because I got around to this newsletter pretty late, I decided that it would be pretty impossible for me to write alot for it, so this month is the month for other people's writings. The major contributor this month is the famous Dan Gazsy, our c99 guru. I think that I'll call him "Mr. C" from now on. Next, I got an article off of Compuserve that is a part of a Multiplan tutorial. This month's section just reviews all of the commands and gives a brief beginners introduction. Then, by Mr. Johan Nykvist, we have our club minutes, and (trumpets please) we have a review, written by Bob Wolff , of VCR Guide.

Here we go!

VCR Movie Guide

A Review by
Bob Wolff

VCR Movie Guide is a very friendly data base program written y Bill Knecht. It is being marketed as a "Fairware" program, hereby the user sends in an appropriate amount of money, based n how much the user likes and uses the program. Bill Knecht of asadena (that's Pasadena TEXAS, not CA) agrees to forward updates nd to feed his puppy if you send him $5.00 if you decide to keep nd use the program.

The program was written to keep track of movies that you have ecorded yourself or bought and keep in a video cassette library. omplete the prompts for each cassette tape (title, rating, assette number, location) and the program can then quickly find our movie with a one word prompt. Try a XXXX rating search.

VCR Movie Guide can store 175 movies in 9 separate program ategories. The search data is displayed on screen. Option 9 ill print the entire data base.

The program comes with two pages of documentation stored as D/V 80 file on the distribution disk.

While sorting and other data handling routines are written in ssembly language, the input prompts are written in BASIC. This llows you to custom modify the program to allow you to keep ecords on practically anything: diskettes, recipes, girl friends, any other information that you can put on-line.

VCR Movie Guide is an efficient program that can quickly rganize your video cassette collection, but the program is not mited to VCRs--since it is written in BASIC, it can be custom dified for almost any sort of data. Definately an "A" rating. d please, $5.00 is a small amount to keep a puppy alive.

CR Movie Guide is available for $2.00 from Dave Green, our nderful club librarian. Please note that this amount ONLY vers copying and disk fees and does not go to the author. ou must pay the author separately!]

MINUTES
January, 1988

The regular monthly meeting of the New Jersey User's Group (NewJUG) was held at the Iselin Public Library on Tuesday evening, 5January 1988.

Mr.Dave Green opened the business portion of the meeting at 7:33 pm. Copies of the minutes of the previous meeting (1December 1987) were distributed to the members without being read.

A MOTION to open nominations from the floor was made seconded and PASSED. As there were no further nominations a MOTION was made seconded and PASSED to close nominations. Since there was a slate of only one candidate for each of the offices, a MOTION was made seconded and PASSED that the election of officers be made unanimous by the Secretary casting a single ballot for their election.

A MOTION was made and seconded to rent a table at TICOFF'88 (East Coast Computer Show), to be used to advertise the club, sell disks, and to promote the working together of the members as a group. TICOFF'88 is to be held on Saturday, 26 March 1988. The table will cost $25. Included in the price of the table are two tickets for admission to the show which may be resold for $5 each. The show will feature the TI99/4A, Myarc Geneve 9640 and various orphans and clones. The table may be used to sell software, used equipment and newsletters. Demonstration may be given of various software, hardware developments, etc. The motion was PASSED.

Two programs were given to members to use and evaluate for the club. A review of the programs are to be included in the next newsletter. The programs will also be demonstrated at our next meeting. Mr.Bob Wolff will review 'VCR Guide' a directory program, and Mr.Dave Green will review 'Side Print'.

Our computer console is being repaired and no demonstration could be given.

Mini reviews of 'XB Tools', 'Fast-Term 116' and 'DiskPrint 2' were given.
Annual Dues for the new year were collected and the meeting was adjou rned at
about 8:50 pm.

Respectfuly submitted,

John I. Nykvist

Johan I. Nykvist
Secretary

QUESTION AND ANSWER TIME

In this column, to appear whenever a few questions arise, I will try to answer said questions as best I can. If I cannot answer the questions, then I am sure that I can find someone who can. So send in your questions!

QUESTION:"How do you pronounce 'Geneve'? I have heard it pronounced about 6 different ways."
ANSWER:"Well, just as accents vary according to where you are raised, so does the pronunciation of this word. But to be most correct, you first need a little background on the origins of the name Geneve. Legend has it that Lou Phillips, mastermind of Myarc (short for Microcomputer Architectures), was stuck for a name for his new computer, and while pacing up and down his stairs, he saw an advertisment poster about a town in Switzerland. Can you guess the name? Yup, the town's name is Geneve! So, to be absolutely correct, you would say it as a native of the town of Geneve does. Now, I haven't been there, so I'm not sure how they say it, so I suggest you do a little 'field research' (a.k.a. vacation) and solve this mystery."

QUESTION:[Question shortened to: why buy a Geneve when there is the lure of the IBM PC and all of its wonders?]
ANSWER:"Well, this is a toughie. To be entirely honest, if you really need all of the power and professionalism of an IBM PC RIGHT NOW, then go get one. The immediate problem with the Geneve is the same as with any other new computer on the market that breaks new ground and that is not compatible with another major computer: a lack of software. Luckily, we have the GPL interpreter which emulates the old TI-99/4A, so we do have some software, but none really that takes advantage of the Geneve's special abilities. If you can wait, then do so, because 1988 is the big year for the Geneve. The year that "will make or break it," because this year Myarc will be concentrating on software, and dozens of other software houses and programmers will be writing Geneve software. Software will have a significant difference from that on the TI, since the Geneve has enhanced grahpics, is faster, and can address more memory. My advice is this: if you need the power and flexibility of an IBM PC type machine right NOW and are willing to possible scrap hundreds (or thousands) of dollars of TI equipment, then your choice must be an IBM PC/XT/AT, etc. But if you can wait a couple of months for software to come out, and you are willing to stick with the TI format then the Geneve is for you.

FOR SALE

Advertisements are free to paid members.

Hardware: 99/4A Console with NEW keyboard (also includes RF mod.)
P.E.B. with a) TI 32K
b) TI RS232
w/ "Y" adapter for RS232 port
c) CORCOMP disk controller card (DSDD)
d) TWO (2) TEAC half-ht disk drives
TI Speech Syn.
Navarone "Widget" (Cartridge expander)
Avatex 1200 bps modem (with cable)
Commodore 1702 13" Composite Color monitor
Panasonic 14" B & W monitor

Furniture: 2 tier computer desk
"Posture" chair (metal framed)

Software: (Cartrides and/or diskettes, with manuals and other accesories)
TI Writer
Editor/Assembler
Microsoft Multiplan
Logo II
Exeltec Extended BASIC
TI FORTH
Personal Record Keeping
Personal Report Generator
Home Financial Decisions
TE II
Munchman
And more...

Printed matter:
All MICROpendium to 2/88
All "99er" magazines
Misc. other TI/99 publications
Books on programming in BASIC and in Assembly.

Asking $800 for the whole package. Will consider selling some pieces separately, but must contact seller with best offer.

Contact: Stan Rosenthal
40 Cottage Place
Gillette, NJ 07933
Evenings: 201 647 2365

DISK SUMMARY
FEBRUARY 2, 1988
59 DISKS TOTAL!

DISKNAME	AVAIL	USED	# PROGS
ANIMATOR	135	223	7
BEST/HYMNS	23	335	24
BEST/SONGS	0	358	18
BITROUTINE	314	44	2
BOARD&PUZL	110	248	7
C-TUTOR:V3	11	347	56
C992*1DOX	294	424	13
C99REL2*1	5	713	46
C99REL3UP	4	354	21
CALENDAR	139	219	9
CATLIB*14	83	275	5
COMP/CRAPS	32	326	9
COMP/UNCOM	322	36	2
DIR99	44	314	12
DM10003/5	4	354	8
DM1000_3:8	24	334	7
DMGR-1000	127	231	6
DSKPRINT	241	117	6
DVUGSAMPLE	4	354	24
FAST-TERM	40	318	13
FASTERM162	147	211	9
FILEREADER	171	187	14
FNLWR3/3-E	573	705	33
FUNL_WRTR	6	352	17
GPL-1	8	350	18
HOMEFINAN	41	317	9
HORIZON-6	0	358	30
MASSCOPY	291	67	3
MENULOADER	156	202	11
MINIBASE99	200	158	5
MXT_4:3	0	358	8
NEATLIST	0	358	16
P-MANUAL	24	334	1
PILOT	0	358	3
PRBASE	630	648	16
PROG_AIDS	205	153	12
SBUG-V3/1+	0	358	7
SCRABBLE	56	302	6
SCREENDUMP	51	307	8
SIDE*PRINT	198	160	6
SIDEWAYS	78	280	16
SIDEWRITER	28	330	11
SPEECHSAVR	78	280	20
STARII	6	352	20
SUPERCOPY	316	42	1
TI-REWRITE	15	343	10
TIMP&TIWRT	67	291	12
TK-WRITER	1	357	17
TRIO+FW1A	226	132	4
TRIVIA99ER	49	309	14
TURBO	319	39	3
UTILITES1	160	198	10
UTILITIES2	211	147	5
VCR/GUIDE	293	65	5
WEATHER	72	286	7
X-MODEM+	108	250	7
XB-IIDEMO	924	514	17
XBTOOLS	63	295	13
X_D	20	338	23

Yes, here is another amazing c99 column by that amazing sysop of an amazing BBS...

c99 and Internal files
by
Dan Gazsy

When Clint Pulley first released the c99 compiler, an individual by the name of Tom Bentley had the foresight to provide library functions for the floating point accumulator (FAC) and a general purpose I/O library called TCIO. Initially I didn't find much of a need for the FLOAT library other then to implement SIEVE type programs.

Two months ago, I decided to write some c programs to perform file maintenance on a bbs message base. My first course of action was to check around to see what c code had been written along these lines. I found the FLOAT and TCIO libraries were all I needed to implement my application. As an added bonus, I ran across a disk catalog function written by Tom Bentley which used both libraries.

Before you write any applications to process internal type files, you should have an understanding of how internal records are structured. When the file is created, you allocate the number of bytes for each record. The fields associated with each record can be either numeric fields or string fields. Numeric fields use 9 bytes; the first byte contains the length of the field (always 8 bytes) used to represent the value in float format. String fields are stored in the same manner as dv80 or df80 strings. The first byte contains the string length; immediately followed by the string. The fields are stored in the record in the sequence they appeared in the print statement (or however the file was created). All unused characters are filled with nulls (0). For example if I created a file of 70 characters and wrote 3 fields (name, address and age) to it, they would look like the following:

```
byte 01    - length of name string
byte 02-10 - Name string ("Dan Gazsy")
byte 11    - length of address string
byte 12-24 - Address string ("22 6th Street")
byte 25    - length of age field (always 8)
byte 26-33 - Age field ("37")
byte 34-70 - Null characters
```

Great you might say! Now how do I read these records from a file into my c program and make decisions based on the values of the fields? Well to start, you open the file with "topen" and read the records with a "tread". In the event you are not sure of the size of the input record, set up an array of 256 bytes to be on the safe side. Once you complete the "tread", the entire record is in the array you specified as the buffer argument. Now all that's left is to parse the record into fields. To do this, you must know how many fields are in each record, what type field they are (string/numeric) and the order they appear in the record. Below appears six functions which we will use to parse the records. The first two functions (getstr and getnum) will be used to parse the input record and the next two functions (putnum and putstr) are used to parse the output record. The last two functions (strlen and getfn) are used to size strings and to open TESTFILE for input or update mode. We lumped all of these into this section because it's what I'd consider a utility. In the interests of good programming practice, we'll compile the following 79 lines of code as you would create a support library. In addition to compiling this program, you'll also have to assemble it. For test purposes, let's call these 79 lines UTILITY;C and the output of the c compiler will be called UTILITY;S. As expected the input file for the Assembler will be called UTILITY;S and the output will be called just plain old UTILITY. Congratulations, you've just created a support library!

```
entry getstr,getnum,putnum,putstr,strlen,getfn;
#include "dsk1.floati"
#include "dsk1.tcioi"

getstr(buff,t)
   int *buff;
   char *t;
   { char *b;
     int j,siz;
     b=*buff;             /* set up b with byte address of input rec */
     j=siz=*b++;          /* set up j & siz with # bytes in string */
     *buff=*buff + j + 1; /* push to next input field */
     while(j--)
       *t++=*b++;         /* transfer string from input rec to string */
     *t = '\000';         /* terminate with null byte */
     return(siz);         /* return size of string */
   }

getnum(buff,t,f)
   int *buff;
   char *t;
   float *f;
   { char *b;
     b=*buff;             /* set up b with byte address of input rec */
     ++b;                 /* point data past the float size indicator */
     *buff = *buff + 9;
     fcpy(b,f);           /* copy the input field into the float */
     ftos(b,t,0,0,0);     /* convert field to string variable */
   }

putstr(buff,t)
   int *buff;
   char *t;
   { char *b;
     int j,siz;
     b=*buff;             /* set up b with byte address of output rec */
     j=siz=strlen(t);     /* set up j & siz with # bytes in string */
     *buff=*buff + j + 1; /* push to next input field */
     *b++=j;              /* put string length in 1st byte of field */
     while(j--)
       *b++=*t++;         /* transfer string to output rec */
     return(siz);         /* return size of string */
   }

putnum(buff,t,f)
   int *buff;
   char *t;
   float *f;
   { char *b;
     int i;
     b=*buff;             /* set up b with byte address of output rec */
     *b++=8;              /* put float id in output rec */
     *buff = *buff + 9;
     while(*t==' ')
       t++;               /* get rid of leading spaces */
     stof(t,b);           /* copy the string field into output record */
   }

```

TESTWRIT;C listed below:

```
extern getfn(),putstr(),putnum(),strlen();
#include "dsk1.tcioi"
#include "dsk1.floati"
#include "dsk1.stdio"

main()
  { int fp,b_ptr,size,i;
    char buff[256],sdum[81],ndum[81];
    float fdum[FLOATLEN];
    putchar(FF);
    b_ptr=buff;
    for(i=0;i<256;i++)
       buff[i]=0;
    while(1)
      { puts("Enter name (max 20 chars):");
        gets(ndum);
        size=strlen(ndum);
        if(size<21)
          break;
      }
    putstr(&b_ptr,ndum);
    while(1)
      { puts("Enter address (max 20 chars):");
        gets(ndum);
        size=strlen(ndum);
        if(size<21)
          break;
      }
    putstr(&b_ptr,ndum);
    while(1)
      { puts("Enter age (max 3 digits):");
        gets(ndum);
        size=strlen(ndum);
        if(size<4)
           break;
      }
    putnum(&b_ptr,ndum,fdum);
    fp=getfn("DSK1.TESTFILE",1);
    if(fp>0)
     {puts("Writing record");
      twrite(buff,0,fp,70);
      tclose(fp);
     }
  }
```

To insure that the files written in c99 are compatible with other languages, we'll read the file and display the fields with a basic program.

```
100 OPEN #1:"DSK1.TESTFILE",RELATIVE,INTERNAL,INPUT,FIXED 70
110 INPUT #1,REC 0:A$,B$,C
120 PRINT "NAME:"&A$
130 PRINT "ADDRESS:"&B$
140 PRINT "AGE:"&STR$(C)
150 CLOSE #1
```

MULTIPLAN TUTORIAL

by

TOM KENNEDY

PART 1: INTRODUCTION TO M.P

ELECTRONIC SPREADSHEETS... CELLULAR REFERENCING... ABSOLUTE REFERENCING... CELLULAR ANALYSIS... WORKSHEETS... FORMULAS...

These are buzzwords of a form of Data Processing that on the surface appears to be incomprehensible to all but accountants and the bridge crew of the Star-Ship Enterprise. As Word Processing is to writing a letter, Data Processing is to using a multiplication table.

Many people have a hard time using spreadsheets, because working with data in this format is similar to learning a new language. But once you learn to use the commands, and the procedure of working with data in a two-dimensional row/column format instead of a one-dimensional equation, you'll find many ways in which Multiplan will allow you to "crunch numbers" faster and easier than using a calculator and notebook. More than that, Multiplan is flexible enough to be used anytime you want to display, or use, numbers or words in a row/column format. In fact, you could even adapt Multiplan to use it as a Word Processor!

What is a spreadsheet? In business, you often hear reference to "our books". The "books" that the businessmen, and we, keep are a pen & paper record showing the Debits and Credits of various bills paid and assets gained, plotted against a scale of time. Each intersection of row and column contains an entry for a value. The last column and/or last row contain a sumation of all previous columns or rows. In an electronic spreadsheet, you recreate the printed form with the addition that each "cell" (a row/column intersection) can also contain a mathematical equation, or "formula", that automatically acts upon pre-defined cells and displays the result accordingly. If any value in any cell is changed, the formula instantly updates displayed results. This self-maintenance ability is what pays off in using an electronic spreadsheet, such as Multiplan.

To operate the Multiplan software on the TI, you must have 32K memory expansion, and at least one disk drive. An RS232 card and a printer are also handy, but not mandatory because unlike word processing, where the end result is a printed piece of paper, the end result with a spreadsheet is useful data, which may be used many ways. Most worksheets are well over 80 characters in width, (up to 2016!) and this requires a cut-and-paste job, so a wide-carriage printer is preferable.

To load Multiplan, you insert the cartridge and program disk, select Multiplan from the menu, and press <ENTER> to load. Before pressing <ENTER> you can select one of eight screen color combinations by pressing the space bar. The first thing you will see is a grid across the top and left side of the screen. These numbers are the row and column locations in the top left, or "HOME" position. There are 255 possible rows and 63 possible columns, with the screen framing a small section. Each "CELL" is referred to by its row/column location, such as: R1C1, R10C22, etc. In R1C1, The Home position, there is a solid rectangle, as large as the width of the cell. This is your cursor, or "CELL POINTER", which is where any entry will appear. Below the cell grid is the COMMAND LINE, which shows the primary commands you will use. There are a number of sub-commands related to each of these, but you must type the first letter of the primary command first, or place the cursor over the command and hit <ENTER>.Below the command line is the MESSAGE LINE, which prompts you for further information when needed. In the bottom left corner is the current cell pointer location, and to the right of that is the contents of the current cell. Lastly, in the bottom right corner is the available memory space remaining.

$$$END OF PART 1, PART 2 NEXT MONTH!$$$

```
strlen(s)
  char *s;
  { int n;
    n=0;
    while(*s++)
      ++n;
    return(n);
  }

getfn(text,m)
  char *text;
  int m;
  { int unit;                /*THESE NEXT LINES WERE BROKEN */
    unit=0;                  /*UP TO PRINT THE NEWSLETTER.ED.*/
    if(m==0)
       unit=topen(&text[0],
                      INPUT+RELATIVE+INTERNAL+FIXED,0);
    else
       unit=topen(&text[0],
                      UPDATE+RELATIVE+INTERNAL+FIXED,0);
    return(unit);
  }

```

To demonstrate that this isn't magic and is compatiable with files created by other programming languages, I've created the input file from a Basic program!

```
100 OPEN #1:"DSK1.TESTFILE",RELATIVE,INTERNAL,OUTPUT,FIXED 70
110 A$="Dan Gazsy"
120 B$="22 6th Street"
130 C=37
140 PRINT #1,REC 0:A$,B$,C
150 CLOSE #1
```

What follows now is simple c program to read the file we created in the above Basic program and display the fields. We will be referencing the library functions we created earlier and code will be included in our main program which will demonstrate how to reference these functions.

TESTREAD;C listed below:

```
extern getfn(),getstr(),getnum();
#include "dsk1.tcioi"
#include "dsk1.floati"
#include "dsk1.stdio"

main()
  { int fp,b_ptr,size,i;
    char buff[256],sdum[81];
    float fdum[FLOATLEN];
    putchar(FF);
    for(i=0;i<256;i++)
       buff[i]=0;
    fp=getfn("DSK1.TESTFILE",0);
    if(fp>0)
```

if(fp>0)

```
       {tread(buff,0,fp,&size);
        b_ptr=buff;
        getstr(&b_ptr,sdum);
        puts("Name:");
        puts(sdum);
        getstr(&b_ptr,sdum);
        puts("\nAddress:");
        puts(sdum);
        getnum(&b_ptr,sdum,fdum);
        puts("\nAge:");
        puts(sdum);
       }
    else
       puts("Error, file could not be opened!");
    tclose(fp);
    puts("\n\nStrike any key to continue");
    getchar();
  }
```

The goal of the function called "main()" has a rather simple purpose. It is to open the file called "TESTFILE", read record 0 into a buffer, parse the buffer into known fields and finally display them. As you can see, I named the program TESTREAD;C, the output code of the compiler was called TESTREAD;S and the assembler output code was named TESTREAD. To execute the program you will have to specify the following files TESTREAD, UTILITY, TCIO, FLOAT and CSUP.

The last part of this column will display a method to combine fields entered on the screen into a record; then write the record back to the file TESTFILE. Like the earlier test, I called this one TESTWRIT;C, the output of the c99 compiler was called TESTWRIT;S and the assembler output was called TESTWRIT. To execute this program you'll need to load the following files: TESTWRIT, UTILITY, TCIO, FLOAT and CSUP. Again remember these are Load and Run (option 3) files.

EAST COAST COMPUTER SHOW - OUR 3RD YEAR

T.I.C.O.F.F.'88

Roselle Park High School

(Exit 137 - Garden State Parkway)

Saturday, March 26, 1988

9:00 - 4:00

All Proceeds go to Student Scholarships!

Admission: $5.00

PRE-PAID ADMISSIONS GET A FREE DISK OF IBM-TI SOFTWARE - SEND CHECK & SASE TO TICOFF
185 W. WEBSTER AVE., ROSELLE PARK, NJ 07204

Huge Indoor Vendors' Area

Workshops - Fairware

Hard/Software - SwapShop

TI-99/4A - MS/DOS Compatibles - I.B.M.

Winner - N.J.A.S.C. TOP 10 PROJECTS AWARD

For Information Call: (201) 241-4550/8902