
Jim Horn, President

Richard D. Sturgell
Bill Whitmore Editors

TI CLUB MEETINGS

THE WASHINGTON DC AREA TI HOME COMPUTER USERS GROUP

The Washington DC Area TI Home Computer Users Group meets monthly at the Fairfax (Va.) High School. The regular meeting night is the second Thursday of each month. The MEETING for FEBRUARY will be held on the regular THURSDAY SCHEDULE at 7:00 PM, February 9th, also the MARCH MEETING will be on Thursday MARCH 8, 1984.

For directions or other info Call Frank Jordan at (301) 899-3882 or Jim Horn at (301) 340 - 9617.

TIBUG - THE BALTIMORE USERS GROUP

The Baltimore Group meets the FIRST TUESDAY of each month at the PINEGROVE ELEMENTARY SCHOOL in Parkville, Md. at 7:00PM.

HAGERSTOWN - WILLIAMSPORT TI USERS GROUP

Meetings are held at the WILLIAMSPORT MEMORIAL LIBRARY on the 3rd FRIDAY of each month. Also a free INSTRUCTIONAL MEETING is held on the FIRST FRIDAY each month. Meetings start at 7:00PM. For more info call Sam Williams at (301) 223-8014.

MONTGOMERY COUNTY TI USERS GROUP

The Group meets at The SLIGO INTERMEDIATE SCHOOL, in the Library at 7:30PM. The regular meeting night is the 4th THURSDAY of each month. For more information call ALLEN MINTON at (301) 493-4502 or DAVE HILL at (703) 941-6876. Either number may be called without toll from Washington DC area. Next meeting THURSDAY, FEBRUARY 23.

BOWIE - CROFTON TI USERS GROUP

The BOWIE - CROFTON Group meets on the THIRD WEDNESDAY of each month. The meetings will be held at the Help Others Help Social Hall in the Bowie Plaza Shopping Center on Rt 197 in Bowie, Md. The meetings start at 7:00PM. For info or directions call Chris Goodman at 262-6633 or 262-5570

SEVERNA PARK TI USERS GROUP

The Severna Park TI Users Group's next meeting will be held at the SEVERNA PARK LIBRARY on McMinsey Rd. near Md. Rt. 2, across from the Severna Park Shopping Center at 7:00PM on FRIDAY, FEBRUARY 3, or MARCH 2, 1984.

TEXAS INSTRUMENTS PROFESSIONAL COMPUTER USERS GROUP

The next meeting in the Washington DC Area of the TI-PC Users Group will be held on TUESDAY, FEBRUARY 14, 1984 at the Computer Age Learning Center classroom, 9431 GEORGIA AVE., SILVER SPRING, MD. at 7:30PM. If you have a PC this will be the meeting that you have been waiting for. For the others that have been trying to decide whether the PC is to be your computer in the near future, you will find the meeting and demos very worthwhile. This meeting room is located only one block from the beltway (I495). Located one block+ south (inside) beltway, on the left side of Georgia Ave. The demo this month will be a hands-on-demo of the new TI Portable Computer.

NINTH ANNUAL TRENTON

COMPUTER FESTIVAL

Bigger & Better Than Ever

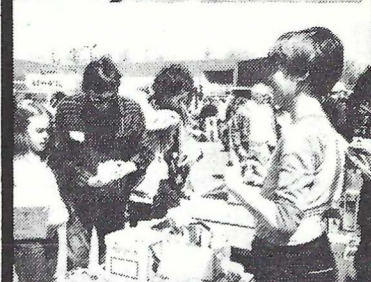
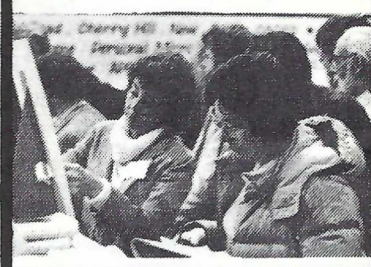
Trenton, New Jersey

SATURDAY, APRIL 14, 1984 10am - 6pm

SUNDAY, APRIL 15, 1984 10 am - 4pm



- Attendance last year exceeded 12,000
- The Original Personal Computer Show
- Huge Outdoor Flea Market—Surplus computer gear, over 5 acres of bargains
- Expanded and Enlarged Indoor Commercial Exhibit Area—rows of booths showing newest products, special discounts
- Forums, Talks and Seminars—Meet the leading experts and hear sessions on Software/Hardware Applications: Robotics, Music, Ham Radio, Educational Computing, (Special sessions for parents and teachers), Business Applications, and New Languages, etc. (Saturday)
- Computer Games Contest—Door Prizes
- Special Conferences—HP41 Conference, ARRL Packet Radio Conference, Educational Computing Conference
- User Groups Meetings—(Sunday) Apple, TRS-80, Commodore, VIC, IBM, S-100, CPM, SIGM Heath, TI, Atari
- Free Short Courses for the general public and novices (Sunday)
- Computer Graphics Theatre
- Day Care Center for Pre-School Children
- Banquet Saturday Night
- and More:

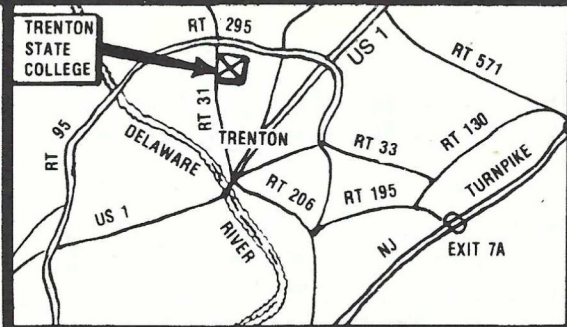


Admission/Registration

- \$ 5 Covers both days and all events (except banquet)
- \$ 3 For students/senior citizens
- \$15 For flea market spot for the entire weekend (includes one admission ticket)

Banquet

- \$12 Saturday night. You may register in advance by sending your check for \$12 per person to:
TCF 84, CN 550, Trenton, NJ 08625



Sponsored by:

Amateur Computer Group of NJ, Philadelphia Area Computer Society, Trenton State College Computer Society, Central New Jersey Computer Club, Institute of Electrical and Electronic Engineers/ACM Princeton Section, Trenton State College Department of Electronic Engineering Technology, TSC Schools of Business, Education and IEET.

Easy to get to:

FREE parking for over 6,000 cars.
Convenient to New York & Philadelphia

For more information contact:

Ms. Marilyn Hughes/Trenton State College/Hillwood Lakes CN 550/Trenton, NJ 08625 (609) 771-2487

CorComp 99/64 Phoenix

UP FROM THE ASHES OF TI...

Apparently, at least one company has recognized the market potential of TI 99/4A related products. A company in California named CorComp is currently marketing 99/4A accessories and will soon market a new computer, the 99/64 Phoenix. On January 11, 1984, I talked to a representative of the company and got some product information to pass on. Please remember that some of the following information is PRELIMINARY and is subject to change.

CorComp's product line concentrates on two areas: TI 99/4A accessories and the 99/64 computer. First, let's talk about 99/4A accessories.

They are presently shipping 32K memory cards that plug into the TI P-Box. Also, they have just started shipping an RS 232 card. Very shortly, they hope to be shipping a disk drive controller card that will support 4 double-sided, double-density drives. This is an improvement over the TI disk controller. All of these products are direct replacements for the TI versions. Prices will be comparable to the price TI was charging for the same accessory.

Also, they plan on marketing an expansion box! If you want to expand your 99/4A, take note! The product will be called an Expansion Chassis and will consist of the following:

1. Four card slots for plugging in accessory cards (TI or third party), such as the 32K, RS 232, Disk Controller, P-Code, CP/M, etc.
2. Power supply large enough to power the accessories plus two half-height drives.
3. Cooling fan.
4. Hardware and signaling connectors for two slimline drives or one full size drive.
5. Cable and interface for 99/4A. Note the connecting cable will be round with a right-angle compact connector to plug in the right side of the console.

The lid for the box is sold separately. You will have three choices here:

1. Plain: This lid simply covers the box and would be used to approximate the TI P-Box.
2. Built-in peripheral lid: It contains 32K expansion memory, an RS 232 interface and the double-density disk drive controller. This option would be used to expand the 99/4A without using accessory cards.
3. 99/64 system lid: This lid contains all the electronics in (2) above, plus the electronics for the 99/64.

Those of you who want to expand your 99/4A now could buy the Expansion Chassis and (1) the plain lid if you have or want to purchase accessory cards, or (2) the lid with built-in peripherals. Later, you could upgrade to the 99/64 by purchasing the 99/64 system lid. Of course, you would have to retire either your accessory cards or the accessory lid. Availability of this product will probably be March or April.

PROGRAM NOTES
By Ken Chandler

EXPANDING THE TI 99/4A

This subject has become the most talked-about one since TI went out of the home computer business. It appears that people are expanding their systems more quickly now, fearing that expansion peripherals will not be available in the future. Judging by surveys taken at our club meetings, the percentage of people with expanded systems has jumped dramatically since October. Also, we have thousands of new owners around who don't know how to put together an expanded system.

The first, obvious, question to ask yourself is, "Should I expand my TI 99/4A or junk it and instead buy one of the other brands, such as Apple, Atari or IBM?" That's a hard question to answer and would depend on what you want to accomplish with a computer and how much money you can spend. However, for general purpose computing, I don't think anyone should casually write off the 4A. I say this because:

1. Graphics: You will be hard pressed to find better all-round graphics on another computer, regardless of price. The Atari comes the closest. Apple and IBM don't have sprites. In fact, you'll find that most accessory graphic boards for other computers use the TMS 9918A Video Display Processor, the same one found in the 99/4A. Several new computers use the same VDP. We have it already!
2. Music: Few of the higher priced models have the musical note-playing ability of the 4A. Also, it's easier to program music on the 4A than any other computer I know of.
3. Accuracy: You'll have to spend thousands more to find a computer that even comes close to the numerical accuracy of the 4A. This accuracy slows it down somewhat, but mathematics is its forte.
4. User friendly: This is where the 99/4A shines! You can't find another computer for less than \$1000 that even comes close. The TI has a lot of built-in subroutines that do difficult chores for you using a single command. You don't have to constantly look up memory addresses to program the 4A.
5. Excellent BASIC: While it is somewhat slower than others, the built-in BASIC easily is the best. Recently, I saw descriptions of the expanded(extra cost) BASIC for the IBM PCjr. and also for the Commodore PET/C64. I was amazed to find that approximately 75 percent of those commands are already part of our console BASIC!
6. Speech: No other computer can provide speech capability as easily and as well as the 4A.

Okay, so you've decided to keep the old 4A for awhile. Why should you expand it? There are many tasks that are impossible to accomplish with the basic system and many that are considerably easier with an expanded system. Much of the more useful, sophisticated software requires extra peripherals. At least 150 of the programs in our own library require something extra(this includes extended BASIC). Saving and loading programs is much easier and faster with a disk drive. You can't print program listings

or letters without the RS 232 peripheral. Extra memory is required for assembly language, word processing, spreadsheet and other programs. Communications over telephone lines requires the RS 232 and a modem.

There are basically two ways to expand the TI 99/4A. You can buy a peripheral expansion box, which will hold the various accessory cards or you can purchase stand-alone peripherals that plug into the right side of the console.

Right now, the only P-box in existence was made by TI. However, they are hard to find. The box contains slots for accessory cards and has space for a single disk drive.

Accessory cards to go in the box are ones made by TI and an ever-increasing number of third-party suppliers. They are:

1. 32K Memory Expansion Card: Provides extra memory and the ability to run assembly-language programs. Requires either extended BASIC, Editor-Assembler or Mini-Memory modules to access it. Cannot be accessed with console BASIC.

2. RS 232 Card: This card has two serial ports and one parallel port. It allows the connection of printers and modems (for communications) to the computer. The computer can access two of these cards.

3. Disk Controller Card: This card is needed to interface 1-3 disk drives with the computer. It will support both single- and double-sided drives. However, it cannot support double-density drives.

4. P-Code Card: This card is needed to run PASCAL and other languages on the 4A. Very few people have this accessory.

5. CP/M Card: This card is third-party developed and allows the 4A to run many CP/M programs. CP/M is a business-oriented language popular on many 8-bit computers.

6. Disk Drive: Most popular brands of disk drives will work with the 4A.

You can also expand the system using stand-alone accessories available from several sources. If you can still find them, the older TI stand-alone peripherals work fine. This method of expansion is attractive because you only need to buy those peripherals you need to accomplish a task. For example, if you only want to print, you can buy a stand-alone RS 232. PERCOM markets a combination disk drive and controller that plugs into the console. However, availability of some of the stand-alone products is limited. Before you buy any non-TI accessory, make sure that the supplier guarantees (in writing) that it is directly compatible with the 99/4A and an exact replacement for the TI accessory. Otherwise, you may find that some software will not execute properly.

You can use a wide variety of printers, disk drives and modems with the 4A. Make sure they are compatible first and also make sure you can get the proper connecting cable.

PROGRAM LIBRARY DIRECTORY

I am working on a comprehensive directory for the club program library. This directory will list each program, category (educational, game, etc.), peripherals required to run it and a brief description of the program and how it works. This is no small task! I got five volunteers to help out at the last club meeting. I could use some more help. Volunteers should preferably have an expanded system

and the portion of the program library they will review. I would prefer not to have to supply programs to the reviewers. If you want to help, contact me at (703)752-9774 (home) or (202)382-8278(work).

SWAP SHOP

I'm sure there must be members out there who have software or hardware to sell or trade. Educational software is one item that becomes obsolete as your children grow older. As a service to club members, future newsletters will dedicate space exclusively for these notices. If you want to sell, trade or buy an item, contact Bill Whitmore at (703)777-2017 or Dick Sturgell at (703)836-1361 or me.

From 3

The 99/64 Phoenix will be a 16-bit, 64K computer that will run all TI software. It will consist of the Expansion Chassis with the 99/64 system lid and a separate keyboard. It will have built-in text-to-speech, real-time clock, FORTH language(maybe), disk drive and can be expanded to one megabyte of memory. Three separate keyboards will be available: a durable child-proof one, one similar to the 99/4A and one similar to the IBM PC. It will have three software cartridge slots: one that accepts TI command modules, a special applications slot for other software adapters and a processor port slot that will enable the 99/64 to run software for other computers, such as Apple, Atari or CP/M. Projected price for all this is in the \$600-700 range! Availability should be second or third quarter of this year, which may be a little optimistic.

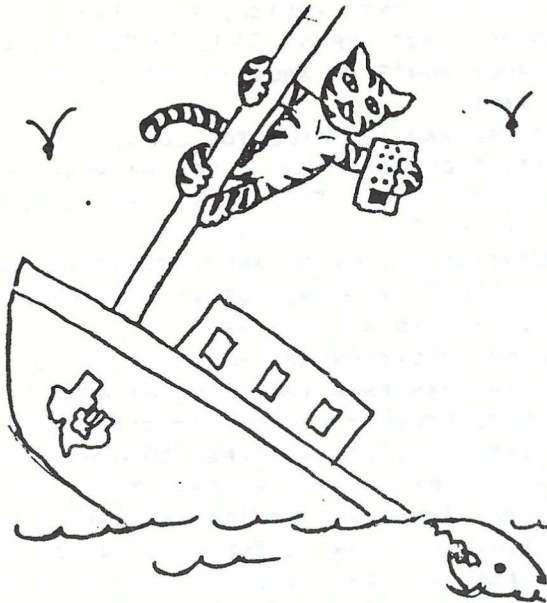
Anyone wanting additional information can contact: CorComp, 23461 Ridge Route Drive, Suite H, Laguna Hills, Ca. 92653

Ken Chandler

LOCAL TI
BULLETIN BOARD!
by Larry Hughes

Yes, it's really true! We now have available to us, a free TI bulletin board -- and whats more, it runs on a 99/4A! It is called TIBBS and is by the famous Ralph Fowler of Atlanta, Georgia. This is the program that TI said couldn't be done! But Ralph did it!!

It is one of the easiest to use boards that I have ever seen. It is available 24 hours a day. It is located in Silver Spring, Maryland and the number is (301) 434-0117. It is provided courtesy of The Softspot, which sells third-party TI software at discount prices. The President of The Softspot and the SYSOP of TIBBS is Phil Simerly. He continually posts the latest TI news and tips and tricks. We thank him for providing this outstanding service to the local TI community!



 * TIGERCUB SOFTWARE *

TIPS FROM THE
TIGERCUB
#7

156 COLLINGWOOD AVE.
 COLUMBUS, OHIO 43213
 (614) 235-3545

125 PROGRAMS IN TI-99/4A BASIC
 ONLY \$3.00 EACH!
 CATALOG \$1.00, REFUNDED ON FIRST ORDER.

THESE ARE NOT PUBLIC DOMAIN PROGRAMS,
 NOT TRANSLATIONS FROM OTHER BASIC LAN-
 GUAGES, NOT PIRATED, NOT MERELY BRIEF
 ROUTINES. THEY ARE ENTIRELY ORIGINAL
 COPYRIGHTED PROGRAMS MAKING FULL USE
 OF THE GRAPHICS, SOUND AND COLOR CAPA-
 BILITIES OF THE TI-99/4A.

NOW ALSO AVAILABLE ON DISKETTE.

THE TIGERCUB WILL NOT DESERT THE SHIP!

HOWEVER, SOME FOLKS SEEM TO BE DETERMINED TO SINK THE SHIP. SOMEONE IS NOW MARKETING A PROGRAM CALLED THE UNPROTECTOR, TO BREAK PROTECTION, AND THE 99'ER HAS SEEN FIT TO ACCEPT ADVERTISING FOR THAT BURGLAR TOOL. AT THE SAME TIME, THE WHIZ KIDS HAVE FOUND AT LEAST THREE DIFFERENT CALL LOADS TO BREAK PROTECTION, AND SOME OF THE USERS GROUP NEWSLETTERS ARE PUBLISHING THEM! EVEN THE COMMAND MODULES MAY NOT BE SAFE - A DEVICE IS BEING MARKETING TO COPY ATARI CARTRIDGES TO TAPE.

SO HOW MUCH LONGER WILL THE PROGRAMMERS BE WILLING TO SPEND MONTHS IN DEVELOPING GOOD PROGRAMS THAT ONE PERSON WILL BUY AND A HUNDRED OTHERS WILL COPY? WHY MUST COMPUTER MANUFACTURERS AND COMPUTER USERS BE SO SELF-DESTRUCTIVE?

THIS DOESN'T AFFECT ME, SINCE I'M PROGRAMMING IN BASIC - I KNOW THAT MY COPYRIGHTED PROGRAMS ARE BEING COPIED AND RECOPIED, EVEN ILLEGALLY SOLD. BUT, UNDER THE CIRCUMSTANCES, I'VE CHANGED MY MIND ABOUT SENDING PROTECTED XBASIC DEMO TAPES OF MY PROGRAMS TO THE USERS GROUP LIBRARIES!

HERE'S A CHALLENGE FOR YOU AMATEUR HACKERS. IN EXTENDED BASIC YOU CAN WRITE IF X=1 THEN Y=7 ELSE IF X=2 THEN Y=33 ELSE IF X=3 THEN Y=19 ELSE IF X=4 THEN Y=21 . IN BASIC THAT WOULD TAKE SEVERAL LINES OF PROGRAMMING - OR WOULD IT? CAN YOU DO IT IN ONE LINE?

SOME OF THE DEEPEST SECRETS OF THE TI ARE BEING DISCOVERED AND PUBLISHED BY CRAIG MILLER IN "THE SMART PROGRAMMER." THEY ARE ALREADY APPEARING ON BULLETIN BOARDS AND ELSEWHERE WITHOUT CREDIT TO CRAIG. PERHAPS THE MOST VALUABLE OF THEM ALL IS:

CALL INIT :: CALL LOAD(-31806,16)

TYPE THAT IN BEFORE YOU BEGIN PROGRAMMING, AND YOU WILL NEVER AGAIN LOSE A PROGRAM TO THAT INFERNAL FCTN = KEY! ALSO, PUT IT IN AS ONE OF THE FIRST PROGRAM LINES AND YOUR PROGRAM WILL BE KIDPROOFED AGAINST THE OPEN-PALM PRESS-ALL KEYBOARD TECHNIQUE.

ANOTHER EXTREMELY USEFUL ONE IS: CALL PEEK(-28672,A) IF THE SPEECH SYNTHESIZER IS ATTACHED, A WILL EQUAL 96, OTHERWISE IT WILL EQUAL 0. IF YOU ARE PUTTING OPTIONAL SPEECH IN AN EXTENDED BASIC PROGRAM, YOU CAN AVOID THOSE SILENT PAUSES BY PUTTING THAT CALL PEEK AT THE BEGINNING OF THE PROGRAM, AND THEN A LINE WITH IF A=0 THEN BEFORE EACH CALL SAY TO SKIP OVER IT IF THE SYNTHESIZER IS NOT ON.

I HAVE ALSO FOUND GOOD USE FOR CRAIG'S ROUTINE CALL INIT :: CALL LOAD(-31888,63,255) . THIS SHUTS DOWN ALL OF THE DISK DRIVE FILES AND MAKES IT POSSIBLE TO LOAD PROGRAMS OVER 12K LONG FROM TAPE, AND COPY THEM BACK TO TAPE, WITHOUT HAVING TO PHYSICALLY DISCONNECT THE DISK DRIVES.

TIPS FROM THE TIGERCUB #7 P.2

I'M NOW RECEIVING ABOUT 40 OF THE USERS' GROUP NEWSLETTERS, AND SOME OF THEM CONTAIN INVALUABLE TIPS THAT DESERVE WIDER CIRCULATION. HERE'S ONE FROM DOUG GERMAN IN THE CENTRAL IOWA 99/4A UG'S "4A FORUM" -

```
100 PRINT "PRESS FIRE BUTTON  
TO CONTINUE"  
110 CALL KEY(1,K1,S)  
120 CALL KEY(2,K2,S)  
130 IF K1#2<>17 THEN 110  
140 JS=INT(K1/18#K2/9+1)
```

NOW, WHEN YOU PROGRAM CALL JOYST(JS,K,S) THE PROGRAM WILL RESPOND TO WHICHEVER OF THE TWO JOYSTICKS YOU WERE HOLDING WHEN YOU PRESSED THE FIRE BUTTON!

IN A PREVIOUS TIPS, I REPORTED THAT IF YOU HAD ABSENT-MINDEDLY TYPED OLD CS1 INSTEAD OF SAVE CS1, AND PRESSED ENTER, YOU CAN SAVE YOUR PROGRAM BY TYPING SHIFT E, ENTER, GET AN I/O ERROR AND START OVER. IT TURNS OUT THAT THIS WILL WORK IN BASIC BUT WILL NOT WORK IN EITHER VERSION OF EXTENDED BASIC UNLESS YOU HAVE THE MEMORY EXPANSION.

I AM NOW THE PROUD OWNER OF A P-BOX, DISK DRIVE, AND MEMORY EXPANSION, SO LET ME PASS ON A FEW THINGS I'VE ALREADY LEARNED THE HARD WAY.

IF YOU HAVE KEYED IN A PROGRAM WITH THE DISK DRIVE TURNED OFF, DO NOT TRY TO TURN IT ON AND SAVE YOUR PROGRAM - YOU WILL LOCK UP THE SYSTEM, WILL HAVE TO SHUT DOWN AND LOSE THE PROGRAM. YOU MUST SAVE THE PROGRAM TO TAPE, THEN TURN ON THE DRIVE, RELOAD THE PROGRAM FROM TAPE AND SAVE IT TO THE DISK.

IF YOU ARE TRYING TO LOAD A PROGRAM FROM TAPE AND YOU GET THE ERROR FOUND IN DATA MESSAGE AT THE VERY BEGINNING, IT MEANS THAT YOUR PROGRAM IS TOO LONG - THE DISK DRIVE HAS STOLEN MORE THAN 2000 BYTES OF YOUR MEMORY. IF YOU JUST WANT TO GET THE PROGRAM INTO THE COMPUTER, YOU CAN DISCONNECT THE DISK DRIVE (IT DOESN'T HELP TO JUST TURN IT OFF), OR USE CRAIG MILLER'S ROUTINE DESCRIBED ON THE PRECEDING PAGE. IF YOU WANT TO

GET THE PROGRAM ONTO A DISK, YOU MIGHT BE ABLE TO DO SO BY FIRST TYPING CALL FILES(1) - BUT YOU PROBABLY WON'T BE ABLE TO GET IT BACK FROM DISK TO TAPE.

IF THE PROGRAM IS STILL TOO LONG, YOU CAN STILL GET IT ONTO A DISK IF YOU HAVE EXTENDED BASIC AND THE MEMORY EXPANSION. THE METHOD RECOMMENDED BY TEXAS INSTRUMENTS INVOLVES DELETING PART OF THE LINES AND THEN RETYPING THEM. MY METHOD IS SLOW, BUT IT ELIMINATES THE RETYPING. PROCEED AS FOLLOWS:

SHUT DOWN OR DISCONNECT ALL DISK DRIVES. LOAD THE PROGRAM FROM TAPE (IN EXTENDED BASIC IF POSSIBLE, BECAUSE YOU CAN DELETE LINES SO MUCH FASTER). DELETE THE FIRST 50 LINES. SAVE THE SHORTENED PROGRAM TO ANOTHER TAPE. TURN ON THE DISK DRIVE, AND MEMORY EXPANSION. LOAD THE SHORTENED PROGRAM IN EXTENDED BASIC. SAVE IT TO A DISK WITH THE MERGE OPTION. SHUT DOWN THE DRIVES AGAIN. LOAD THE ORIGINAL PROGRAM FROM TAPE. DELETE THE LAST 50 LINES. SAVE IT TO A TAPE. TURN ON THE DISK DRIVE AND MEMORY EXPANSION. LOAD IT BACK IN, IN EXTENDED BASIC. USE THE MERGE COMMAND TO MERGE THE PREVIOUS SHORTENED PROGRAM INTO THIS ONE. SAVE THE MERGED PROGRAM TO A DISK. IT WILL SHOW UP ON THE CATALOG AS A FILE, BUT WILL LOAD AND RUN AS A PROGRAM. DON'T GIVE IT TO ANYONE WHO DOES NOT HAVE EXTENDED BASIC AND MEMORY EXPANSION!

ACCORDING TO DUANE FISCHER IN THE 99/4 USERS OF AMERICA NEWSLETTER, THERE ARE TWO MODELS OF THE NEW WHITE-PLASTIC TI-99/4A. THEY CAN ONLY BE IDENTIFIED BY THE COPYRIGHT NOTICE ON THE BOTTOM RIGHT EDGE OF THE TITLE SCREEN. IT WILL READ EITHER (C) TEXAS INSTRUMENTS 1981 OR (C) TEXAS INSTRUMENTS V2.2 - AND IF IT IS THE LATTER, IT IS ONE OF THOSE WHICH WERE GIMMICKED SO THEY WOULD NOT RUN THE COMMAND MODULES PRODUCED BY FUNWARE, ROMOX, ATARI OR PARKER BROS.

IF YOU ARE HAVING TROUBLE DETERMINING LOCATIONS ON THE SCREEN DURING PROGRAMMING, PUT ON A TEMPORARY GRID WITH:

```
1 CALL CHAR(32,"FF80808080808080")
```

IF IT WON'T INTERFERE WITH YOUR GRAPHICS,

YOU CAN EVEN NUMBER THE GRID:

```
2 FOR J=1 TO 32  
3 C=C+1-ABS(C#10)  
4 CALL HCHAR(24,J,48+C)  
5 IF J 24 THEN 7  
6 CALL VCHAR(J,14,48+C)  
7 NEXT J
```

ALMOST OUT OF MEMORY, SO HAPPY HACKIN'
JIM PETERSON.

NOTES FROM A NOVICE
By Penn James

QUESTION: Do you have to be a programmer to enjoy the full use of your TI-99/4A home computer?

Millions of ordinary folks now own home computers. However, a recent survey by LINK of New York concludes that one out of four home computers is gathering dust in the closet. What is astounding though is the revelation that close to 60% of those in use are used almost exclusively for games and entertainment. This leaves less than one out of five being utilized for "serious computing". What is the reason for this state of affairs? I am sure the reasons are many and varied. But one significant reason, I suggest, is what I call the programming myth.

While not knocking programming or programmers, I do contend that knowledge of programming is NOT ESSENTIAL to gaining complete control of your home computer to enable you to enjoy full use of its various capabilities. In fact, knowledge of programming does you no use with such programs as TI-Writer and Multiplan. You cannot change or even list the programs in the command modules. But you can learn to use these programs, to their full measure of power without ever learning to write one line of BASIC!

In this installment, I would like to discuss my experiences and development with the Personal Record Keeping command module--sometimes referred to simply as PRK or PRK module. In future installments, I should like to cover other non-game programs you can easily learn to master and enjoy.

Once you understand a few basic concepts about the organization of the PRK module, the uses to which you can put it are limited only by your imagination. The line between the "documentation"(instruction book)and a magazine tutorial(allegedly an article on "how to use it")is often blurred by the fact that the tutorial "explains" the "how to" part in exactly the same unfamiliar terms as the documentation! Please note that this tutorial is NOT designed to relace the need to read the documentation which comes with the PRK module. It is intended to supplement the instructions booklet and to be considered in reference to the instruction book.

I think most writers are fearful of writing plain English lest it be thought they do not know what they are talking about. I have no such fear because I know that I don't know what I am talking about. But I know what I've done with it and that's what I aim to tell you about.

First, think of the PRK module as an unlimited stack of notebooks all with blank pages. The number of pages in each book will vary according to the structure that you will create with each book as you "take" it out of the "stack" and use it. Each page will contain the amount of writing space you will choose when you create the structure of your book in step-by-step prompting on your screen. Once you choose the amount of writing space in the structure, each page in the book will have the same amount of space. You will have one opportunity, at the beginning, to change it; after that it is fixed and can only be changed by "throwing away" the book and starting a new one. Remember: you have an unlimited supply, so don't worry about running out of books. The computer reserves the writing spaces you choose, calculates the total for you and divides it into the available memory, then tells you how many pages you will have in your book. The computer does this automatically without any further instructions from you. NOTE: The less space you use per page will give you more pages in your book. Conversely, the more spaces per page used, the less number of pages available to you in the book.

Since the spaces on the page are limited, you will only record one event per page. For example: you write check number 110 to the Telephone Co. for \$25.00 on Feb. 14, 1984; although there are four items of data (information) here, it is still only one event or related unit. The four data items are (1) the check number, (2) the payee, (3) the amount, and (4) the date. Of course, we could add more data items if we like, depending on what information we want to record about the event.

When first starting to work with your PRK it is best to plan your "notebook" pages by drawing the structure on paper. Later, it will become second nature to you and you can easily design your notebook pages at the keyboard. Let's design a notebook to keep track of an imaginary beer can collection. We will call the notebook "BEERCANS"--which now becomes our FILENAME, whenever the screen prompts you to input the filename. Now take a piece of blank, lined paper and turn it sideways. Draw perpendicular lines to make five or six columns. Now, let's think about the kind of information we would want to record for our collection. First, we would probably want to list the cans by Brand--so let's label the first column BRAND; next, we might want to know each can's value--so let's label column 2 as VALUE; in column 3 we might want to show the date we sold it, so let's label that column as DATE SOLD. You could label the other columns, but for brevity let's stop with three data items.

Before we create this file in the computer, there is one more thing we want to do. And that is to designate the number of spaces each page is to have. On your paper, number some lines down the left hand margin from 1 to 10. In the computer, they would be the same thing as "pages" 1 to 10. Now, on line 1, column 1, write BREW MEISTER; in column 2, same line 1, write .25 (meaning 25 cents); in column 3, write 12/31/83. On line 2, in column 1, write BULL MALT; in column 2, write 1.00; leave column 3 blank. Now look at it. Now you have an idea of how your screen or print-out will look after you have entered the so-called "raw" data items.

If you use 15 character spaces for column one, 8 decimal spaces for column two, and 8 character spaces for column three, then you will have 414 pages available to you in your "BEERCAN" file. That simply means that you have enough space to record the brand name, value, and date sold for 414 cans--one can (event, item, transaction) per page. To illustrate a point made earlier, if you change the 15 character spaces in column one to 13, then you will increase the number of pages to 446. And that applies to any of the columns. Increase any of the column spaces and number of pages decreases; decrease any of the column spaces and the number of pages increases. Why is that important? Because, with a little planning and wise use of column space you can control pretty accurately the number of pages available to you in relation to your needs.

A note of caution is in order on "filling" up your electronic notebook. In order to assure that you do not get a "MEMORY FULL IN --" message when you later LOAD your file to work with it, do not use every "page" in the "book". The rule of thumb is to subtract 2% of the available pages. Thus, if a file contains 100 pages, use only 98 of them. In our beer can example of 414 pages, we would subtract 8 or 9 pages and use only 405 pages of the file.

Your PRK is in reality a mini-spreadsheet: You can design columns and rows to fit almost any situation and condition. The more you work with it, the more things you will think up to do with it. It is an amazing program and ten times more valuable than its current depressed price. Pick one up, and you'll get hooked on using your computer more than for just games. BYE.
1/84

CONVERSION FROM TRS-80 BASIC AND APPLESOFT TO TI-BASIC.

The following is an article that appeared in TISOFT, the newsletter of the Belgian TI club, Selstbaan 24, B-2080 KAPELLEN, Belgium. The club started about four years ago, concerning itself mainly with the SR-52 and the TI-59 programmable calculators. Now they have almost completely switched over to the TI-99/4A. The club is run by four high school math teachers. Besides the newsletter, the club has a program exchange with over 500 programs. Many of their programs are in several languages, such as Dutch, French, German, etc. Some of their programs, after suitable translation, have found their way into our own club library, although you might not be

aware of it yet.

The TISOFT club has close to 600 members, which is astonishing, as Belgium is only about the size of Maryland. But it has about 11 million inhabitants. Most of the people speak Flemish, a language closely related to Dutch and to English. But more than 4 million people still speak French and even a few hundred thousand use German as their mother tongue. To escape this Tower of Babel, Thomas Coppens, the club president, decided to use English for his newsletter, satisfying this way all the factions and making his newsletter more accessible to the rest of Europe.

Maurice E.T. Swinnen

SOFTWARE CONVERSION

All of us microcomputer programmers probably read some magazines about programming. Most of the time these magazines contain programs. The problem is that most of the encountered programs are not written in TI-Basic. In that case we are forced to rewrite these programs in TI-Basic.

The intention of this article is to help our readers in converting programs written in TRS-80 (Microsoft) BASIC or Applesoft BASIC. The difficulty depends for a great part in the character of the program to be converted.

If the program has no or very few graphics the task is most of the time an easy one. If there is heavy use of graphics you should expect to rewrite it.

I. TRS-80 BASIC to TI-BASIC

We will give a summary of commands and the way of converting them (if possible). After that we will give a few examples.

a) TRS-80 Commands not requiring conversion

ABS	ASC	ATN	CHR\$
COS	DATA	DIM	END
EXP	GOTO	GOSUB	INPUT
INT	LEN	LET	LOG
ON/GOSUB	ON/GOTO	PRINT	READ
REM	RESTORE	RETURN	SGN
SIN	SQR	STR\$	TAN
VAL			

b) TRS-80 Commands easily converted

<u>TRS-80</u>	<u>TI BASIC</u>
CLS	CALL CLEAR
FIX	INT
INKEY\$	CALL KEY
LEFT\$(A\$,N)	SEG(A\$,1,N)
MID\$(A\$,N1,N2)	SEG\$(A\$,N1,N2)
RIGHT\$(A\$,N)	SEG\$(A\$,LEN(A\$)-N+1,N)
RANDOM	RANDOMIZE
RND(N)	INT(N*RND+1)
STOP	BREAK
TAB	TAB, (with comma)
?	PRINT
!	REM

c) TRS-80 Commands that can be ignored

CLEAR	CSNG	DEFSGN	CDBL	DEFDBL
DEFSTR	CINT	DEFINT	FRE	

d) Commands difficult to convert to TI BASIC

TRS-80

TI BASIC

IF .. THEN .. ELSE

IF .. THEN .. ELSE (!)
refer to line numbers

POINT

CALL CHAR CALL GCHAR

POKE (graphics)

CALL CHAR CALL HCHAR

PRINT AT

FOR .. ASC .. CALL HCHAR..
NEXT

PRINT .. FOR .. PRINT " " ..
NEXT

RESET

CALL CHAR CALL HCHAR

SET

CALL CHAR CALL HCHAR

(!) no problem in TI Extended BASIC

e) Commands not available in TI BASIC (!!)

ERL ERR ERROR ON ERROR

PEEK POKE POS RESUME

STRING\$ USR VARPTR PRINT USING

(!!) Most of the equivalent commands are available in
TI Extended BASIC

Some examples:

1. One of the first problems one encounters is the conversion of the PRINT AT (ampersand really). This command makes it possible to print a message or a value everywhere on the videoscreen. This is rather difficult in TI Basic since the printing always begins on the last line of the screen. When continuing the printing process the screen scrolls up. If you don't mind the scrolling then a possible solution is:

```
100 PRINT "MESSAGE AT THE MIDDLE OF THE SCREEN"  
200 PRINT :::::::::::
```

This gives a quick scroll-up to the center of the screen. Another possible solution is setting up a loop as follows:

```
100 A$="SOMETHING TO BE PRINTED STARTING AT 10,12"  
200 FOR I=1 TO LEN(A$)  
300 N1=ASC(SEG$(A$,I,1))  
400 CALL HCHAR(10,12,N1)  
500 NEXT I
```

This works fine in theory but it is very slow when the string

is long.

2. Suppose the following TRS-80 program portion:

```
120 FOR I1=1 TO N1
130 IF IZ(I1)<>0 THEN PRINT "THIS ITEM DROPPED";ID$:GOTO 160
140 IF K$=K1$ THEN IF C1(I1)=0 GOTO 160 ELSE C2=10*C1(I1)+200
150 DX=C3-C2
160 NEXT
```

The converted program could be something like this:

```
120 FOR I1=1 TO N1
130 IF IZ(I1)=0 THEN 140
132 PRINT "THIS ITEM DROPPED";ID$
134 GOTO 160
140 IF K$<>K1$ THEN 150
142 IF C1(I1)=0 THEN 160
144 C2=10*C1(I1)+200
150 DX=C3-C2
160 NEXT I1
```

We hope this conversion table and the few examples given will help you in successfully converting TRS-80 programs and make them run on your beloved TI.

II. Applesoft to TI-BASIC

We will not deal with Applesoft graphics in this article. First of all the APPLESOFT language card has about 29 non-graphic commands which are identical to TI BASIC. They are shown in the table below and can be copied without much concern over compatibility.

ABS	ASC	ATN	CHR\$
COS	DATA	DEF	DIM
END	EXP	FOR..TO	GOSUB
GOTO	INT	LEN	LET
LOG	ON..GOTO	ON..GOSUB	READ
REM	RETURN	SGN	SIN
SQR	STEP	STOP	STR\$
TAN			

In the remaining 26 or so commands the differences vary from very slight to major. Most importantly the differences though slight in format or content can cause major problems in converting code.

We'll look over each command and try to resolve difficulties:

a) String Commands

Since Applesoft BASIC uses the same commands LEFT\$, MID\$ and RIGHT\$ as does TRS-80 Basic the conversion problem is the same.

The VAL function acts the same way in both APPLESOFT and TI BASIC if the field being VAled is a valid numeric string. That is, both will return 45.2 as the value of "45.2". If the string does not contain valid numeric characters, however, the results are very different. TI BASIC will stop the program if the field contains non-numeric characters. APPLESOFT, however, will return with the numeric equivalent of the numbers found in the string before the first non-numeric character.

For example: VAL("123AB") will return with 123 in APPLESOFT. If the first character is non-numeric then APPLESOFT returns a \emptyset .

This is important because it means that APPLESOFT does not have to edit a string prior to the VAL statement.

A typical program will have code such as:

```
1 $\emptyset$  INPUT A$
2 $\emptyset$  X=VAL(A$)
3 $\emptyset$  IF X= $\emptyset$  THEN 1 $\emptyset$ 
```

In most cases this can be replaced by INPUT X.

If this is not possible the replacement could be as follows:

```
1 $\emptyset$  FOR Y=1 to LEN(A$)
2 $\emptyset$  IF (ASC(SEG$(A$,Y,1))<48)+(ASC(SEG$(A$,Y,1))
>57) THEN 4 $\emptyset$ 
3 $\emptyset$  NEXT Y
4 $\emptyset$  Y=Y-1
5 $\emptyset$  Y= VAL(SEG$(A$,1,Y))
```

This is not a rigorous equivalent of APPLESOFT's VAL but is sufficient for whole numbers greater than -1.

b) FOR-TO-NEXT-STEP

In the usual run of programs there is no problem. But beware in the following case:

The statement FOR Z=5 TO 4 will execute once in APPLESOFT (in TRS-80 BASIC too) but will not execute at all in TI BASIC. This can easily be seen in a program if the numeric values are like in the example. It is not so obvious if there are variables like in:

```
FOR Z=A TO B
```

The safest thing to do is to test A and B, and if A is greater

then make A equal to B before entering the loop.
NEXT gives no problem except that in APPLESOFT it may be used without variable in a single loop. This is not the case in TI BASIC.

c) INPUT/OUTPUT

APPLESOFT gives the possibility to print at any point of the screen by using VTAB and HTAB commands.

Here the same remarks can be made as for TRS-80 BASIC.

The APPLESOFT GET instruction can be replaced by CALL KEY.

d) SCREEN Commands

The APPLE has three modes of processing: TEXT mode and two graphic modes.

While in TEXT mode there is a great variety in screen control commands.

It is indeed possible to control the text window.

Commands are:

CALL -936 clear the screen inside the text window

CALL -912 scrolls the text window up one line

CALL -868 clears the current line from the cursor to the right

HOME same as TI's CALL CLEAR

POKE 32,L sets left margin of window to L

POKE 33,W sets width of window

POKE 34,T sets top of window

POKE 35,B sets bottom of screen

FLASH start 'flashing' output from white letters on black to black letters on white and back again

INVERSE reverses output to black letters on white

NORMAL resets FLASH and INVERSE

POS(N) get current horizontal column of cursor (i.e. N will have column number 0-39)

To simulate FLASH or INVERSE, use the COLOR statement of TI BASIC. For example, CALL COLOR(3,16,2) gives white numbers from 0 to 7 on a black background. Changing this to CALL COLOR(3,2,16) will cause the inverse of it to appear.

e) Random Numbers

APPLESOFT's ability to retain a random number for re-use can be confusing.

If the statement is RND(0) APPLESOFT reuses its last random number. If the statement is RND(N) where N is positive it gives a new random number. If N is negative N acts as a seed number and all other RND statements will follow a standard sequence.

If you see a statement using RND(\emptyset), backtrack to the last statement with RND(N) and save that random number in place of RND(\emptyset). For example:

```
1 $\emptyset$  IF RND(2) > .5 THEN 5 $\emptyset\emptyset$ 
.
.
6 $\emptyset$  IF RND( $\emptyset$ ) > .75 THEN 6 $\emptyset\emptyset$ 
```

would be converted to:

```
1 $\emptyset$  Q=RND
15 IF Q > .5 THEN 5 $\emptyset\emptyset$ 
.
.
6 $\emptyset$  IF Q > .75 THEN 6 $\emptyset\emptyset$ 
```

f) IF-THEN-ELSE

APPLESOFT does not require the ELSE feature of an IF statement because it allows other statements to follow the THEN part of the IF statement. The following shows how this works:

```
1 $\emptyset$  IF X=A THEN X=X+1:Y=Y+1.
2 $\emptyset$  A=X+Y
```

If X is equal to A all statements following the THEN are executed. If X isn't equal to A, the program simply advances to statement 2 \emptyset . The TI BASIC equivalent is:

```
1 $\emptyset$  IF X=A THEN 15 ELSE 2 $\emptyset$ 
15 X=X+1
16 Y=Y+1
2 $\emptyset$  A=X+Y
```

One result of multiple statement lines is that TI programs require much more coding and a concurrent increase in memory needed for code. Keep this in mind if you are tempted to enter a program requiring 16K RAM in APPLESOFT. It probably won't fit in your TI machine! (If you have Extended Basic, there will be no problem since this Command Module allows multiple statement lines!!).

g) Logical Expressions

Both interpreters allow logical expressions to be used as if they were numeric values. APPLESOFT treats true expressions as if they are equal to 1, while false expressions are equal to \emptyset . For TI BASIC true expressions are -1, false are \emptyset . Whenever converting code from APPLESOFT, just insert a "-" in front of the logical expression:

```
10 X=(O$="A")*5 .
```

becomes

```
10 X=-(O$="A")*5
```

h) AND/OR

APPLESOFT allows multiple IF tests to be combined using the BOOLEAN operators AND and OR. TI BASIC also allows this using the "*" and "+" arithmetic expressions respectively. For example:

```
1Ø IF (A=B) AND (C=D) THEN X=X+1
```

is replaced with:

```
1Ø IF (A=B)*(C=D) THEN 15 ELSE ...
15 X=X+1
```

In some cases a straight conversion of the APPLESOFT IF..THEN will result in wasteful code. It is always a good idea to understand the purpose of the tests being made, and if possible re-code them more efficiently. For example:

```
1Ø IF (A=B) AND (C=D) THEN X=X+1
2Ø Y=Y+1
```

would convert to:

```
1Ø IF (A=B)*(C=D) THEN 15 ELSE 2Ø
15 X=X+1
2Ø Y=Y+1
```

but it would take less code (less memory!) to invert the test:

```
1Ø IF (A<>B)+(C<>D) THEN 2Ø
15 X=X+1
2Ø Y=Y+1
```

i) Special Functions

CLEAR : initializes all variables. Automatically done by TI BASIC as part of 'RUN'ning

HIMEM : sets highest and lowest memory address available to BASIC. No equivalent in TI BASIC.

FRE(Ø) : gets amount of available memory left.

As can be seen from the foregoing most of the conversion is straightforward with most of the effort devoted to convert PRINT statements.

The most important: don't get frustrated if your first attempts don't succeed the way you intended. After a while things will go a lot easier.

S O R T I N G .

The following is an article that appeared in TISOFT, the newsletter of the Belgian 99-er club. It is written by the editor Thomas Coppens. The article presents practical sort routines and rates each one with respect to speed of execution. A final demonstration program contains all of the routines. A table shows comparisons among the various sorting methods, each one used with 10, 20, 50, 100, 200 and 400 elements.

Maurice E.T. Swinnen

An important programming problem is the sorting of numbers or strings. This means ordering the numbers (strings) in ascending or descending order (ascending or descending alphabetical order).

In this article series we will try to give you a number of methods for sorting which can be programmed in BASIC. We will also try to see how efficient (this means quick!) each method is.

Therefore we will always sort the same array of numbers and give the time needed. By the way, you can check the time for yourself!!

1. BUBBLE SORT

In this rather simple method one goes through the row of numbers for example from left to right. When two adjacent numbers are in the wrong order (first bigger than second when ordering in ascending order) they are swapped. This procedure is repeated until no more swapping has to be done.

The advantages are: simple, short program.

The disadvantages: much swapping, much 'loop'-work and slow.

Here follows the program:

```
10 REM BUBBLE SORT
20 DIM A(100)
30 RANDOMIZE (2)
100 REM SET UP RANDOM ARRAY
110 FOR I=1 TO 100
120 A(I)=INT(RND*99999)
130 NEXT I
135 PRINT "START"
140 REM START OF SORT
150 FOR I=2 TO 100
160 IF A(I)>=A(I-1) THEN 200
165 REM SWAP THEM
170 X=A(I)
180 A(I)=A(I-1)
190 A(I-1)=X
195 F=1
200 NEXT I
205 REM WAS THERE A SWAP?
210 IF F=0 THEN 260
220 F=0
230 GOTO 150
250 REM END OF SORT
260 PRINT "END"
265 REM PRINT SORTED ARRAY
270 FOR I=1 TO 100
280 PRINT A(I)
290 NEXT I
300 END
```

```
10 REM INTEGER FLAG SORT
20 DIM A(100)
30 RANDOMIZE (2)
100 REM SET UP RANDOM ARRAY
110 FOR I=1 TO 100
120 A(I)=INT(RND*99999)
130 NEXT I
135 PRINT "START"
140 REM START OF SORT
150 N=100
160 R=N
170 F=0
180 FOR I=1 TO R-1
190 IF A(I)<=A(I+1) THEN 215
195 X=A(I)
200 A(I)=A(I+1)
210 A(I+1)=X
214 F=I
215 NEXT I
220 IF F=0 THEN 260
225 R=F
230 GOTO 170
250 REM END OF SORT
260 PRINT "END"
265 REM PRINT SORTED ARRAY
270 FOR I=1 TO 100
280 PRINT A(I)
290 NEXT I
300 END
```

The approximate times for sorting 100 elements are:

Bubble Sort : 3'20"

Integer Flag Sort : 2'31"

2. SHAKERSORT

When doing a bubblesort by hand on paper to see what happens, one remarks that the biggest element is fast bubbling up (after one loop it is in its place). This is not the case, however, for the smallest element. This problem is taken care of in the so-called Shakersort. In fact this is a bubble sort that goes both ways. One time the loop goes from left to right, the next time from right to left. This makes the name obvious. It may be clear that this can be combined with all the special features for the amelioration of the bubblesort.

The program can be found on the next page.

```

100 REM SHAKER SORT
110 DIM A(100)
120 RANDOMIZE (2)
130 REM SET UP RANDOM ARRAY
140 FOR I=1 TO 100
150 A(I)=INT(RND*99999)
160 NEXT I
170 PRINT "START"
180 REM START OF SORT
190 N=100
200 L=1
210 R=N
220 F=0
230 FOR I=L TO R-1
240 IF A(I)<=A(I+1) THEN 290
250 X=A(I)
260 A(I)=A(I+1)
270 A(I+1)=X
280 F=1
290 NEXT I
300 IF F=0 THEN 460

310 R=R-1
320 IF R=L THEN 460
330 F=0
340 FOR I=R TO L+1 STEP -1
350 IF A(I)>=A(I-1) THEN 400
360 X=A(I)
370 A(I)=A(I-1)
380 A(I-1)=X
390 F=1
400 NEXT I
410 IF F=0 THEN 460
420 L=L+1
430 IF L=R THEN 460
440 GOTO 220
450 REM END OF SORT
460 PRINT "END"
470 REM PRINT SORTED ARRAY
480 FOR I=1 TO 100
490 PRINT A(I)
500 NEXT I
510 END

```

The time needed to sort the same 100 elements is 2'20".

3. SWAPSORT

This is a different kind of sort. We begin looking for the smallest element of the list. This element then comes in the place of the first element. Then we look for the second smallest element, which replaces the second element, and so on.

The program listing is:

```

10 REM SWAPSORT
20 DIM A(100)
30 RANDOMIZE (2)
100 REM SET UP RANDOM ARRAY
110 FOR I=1 TO 100
120 A(I)=INT(RND*99999)
130 NEXT I
135 PRINT "START"
140 REM START OF SORT
145 N=100
150 FOR I=1 TO N-1
155 R=I
160 FOR J=I+1 TO N
170 IF A(R)<=A(J) THEN 190
180 R=J
190 NEXT J
195 IF R=I THEN 230
200 X=A(I)
210 A(I)=A(R)
220 A(R)=X
230 NEXT I
250 REM END OF SORT
260 PRINT "END"
265 REM PRINT SORTED ARRAY
270 FOR I=1 TO 100
280 PRINT A(I)
290 NEXT I
300 END

```

The time for sorting the 100 numbers is 1' 05".

4. SHELLSORT

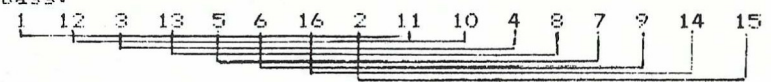
The Shellsort is in fact an enhancement of the bubble sort. The idea behind this sort is to switch numbers over a larger distance first and then refine. Let's take an example:

1 12 3 13 5 6 16 2 11 10 4 8 7 9 14 15

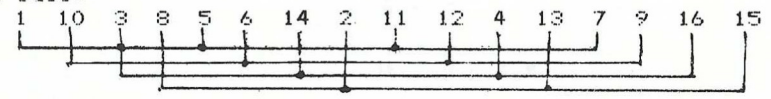
Suppose we want to sort this list of 16 numbers. In this case the sort begins with 8 groups of 2 numbers. These groups are sorted by one of the already mentioned methods. Then the list is divided in 4 groups of 4 numbers and each set is ordered. In this manner the algorithm continues until we have one list of 16 numbers. After this list is sorted the work is finished.

Let's take a look at what happens in the different passes of the sort:

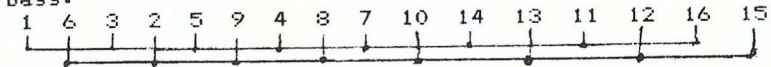
First pass:



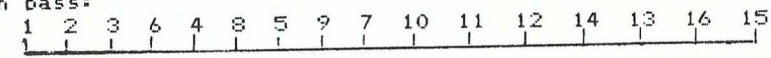
Second pass:



Third pass:



Fourth pass:



Now the last sort takes place. This sort is a very good type of sort to use in programs. It is not too difficult to program in BASIC. Here is a program of the Shellsort:

```

10 REM SHELLSORT
20 DIM A(100)
30 RANDOMIZE (2)
100 REM SET UP RANDOM ARRAY
110 FOR I=1 TO 100
120 A(I)=INT(RND*99999)
130 NEXT I
135 PRINT "START"
140 REM START OF SORT
145 N=100
170 M=N
175 M=INT(M/2)
180 IF M=0 THEN 270
185 FOR ST=1 TO M
190 I=ST
195 J=ST+M
200 F=0
205 IF A(I)<=A(J) THEN 235
210 F=1
215 X=A(I)
220 A(I)=A(J)
225 A(J)=X
235 I=J
240 J=J+M
245 IF J=N THEN 205
250 IF F=1 THEN 190
255 NEXT ST
260 GOTO 175
265 REM PRINT SORTED ARRAY
270 REM END OF SORT
275 PRINT "END"
280 FOR I=1 TO 100
285 PRINT A(I)
290 NEXT I
300 END

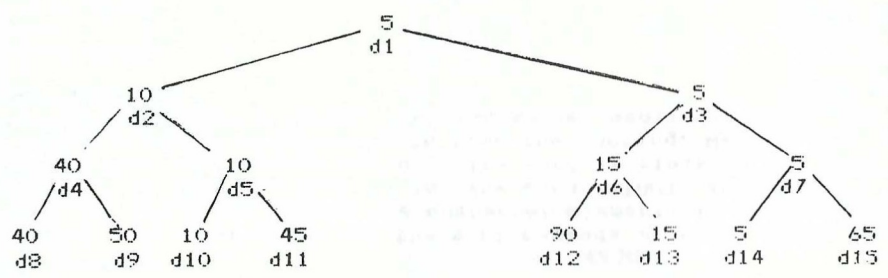
```

5. HEAPSORT

This is an entirely different kind of sort. Here the sorting is done by building binary trees. A tree is an arrangement of elements as shown beneath. Take for example the numbers:

40 50 10 45 90 15 5 65

When ordering these numbers in a tree so that the smallest number is used as root, one needs N-1 comparisons.



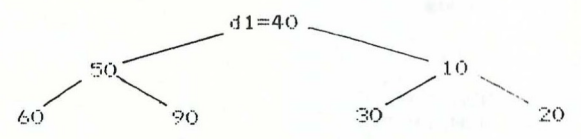
When such a binary tree has to be stored we have the following condition:

$$d(i) \leq d(2i) \text{ and } d(i) \leq d(2i+1)$$

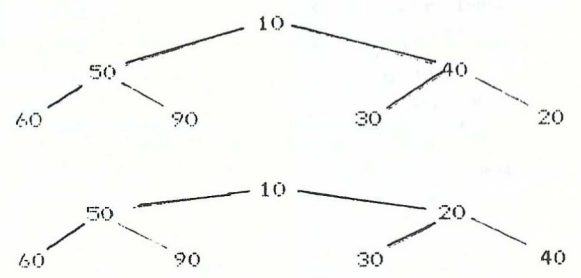
An ordering in that way is called HEAP.

Suppose now that the head of the following figure has to take one more element (d1=40).

21



The new element is placed at the head of the heap. Then the element walks down along the smaller elements and at the same time the small elements go upwards.



So the method of the heapsort is as follows:

- building of the heap
- working out the heap

A listing of the heapsort follows:

```

10 REM      HEAPSORT
20 DIM A(100)
30 RANDOMIZE (2)
40 REM      SET UP RANDOM ARRAY
50 FOR I=1 TO 100
60 A(I)=INT(RND*99999)
70 NEXT I
80 PRINT "START"
90 REM      START OF SORT
100 N=100
110 L=INT(N/2)+1
120 M=N
130 IF L=1 THEN 170
140 L=L-1
150 X=A(L)
160 GOTO 210
170 X=A(M)
180 A(M)=A(L)
190 M=M-1
200 IF M=1 THEN 330
210 J=L
220 I=J
230 J=J*2
240 IF J=M THEN 280
250 IF J>M THEN 310
260 IF A(J)>=A(J+1) THEN 280
270 J=J+1
280 IF X>A(J) THEN 310
290 A(I)=A(J)
300 GOTO 220
310 A(I)=X
320 GOTO 130
330 A(1)=X
331 PRINT "END"
340 FOR I=1 TO N
350 PRINT A(I)
360 NEXT I
370 END
    
```

6. QUICKSORT

This is probably the fastest sort.

The procedure is as follows:

- choose an arbitrary element from the array to be sorted. For example: the element in the middle.
- search the array from the left and from the right to the middle. Do this until you find coming from the left an element larger than the chosen and coming from the right an element smaller than the chosen element. Swap these elements and continue the procedure until the pointers meet each other. At that moment the array is split in two, where the elements in the left part are smaller than the elements in the right part.
- now sort each part of the array as mentioned above until the parts are formed by one element.

The algorithm is straightforward but the BASIC program is not that easy. For each partition the left and right ends have to be put in memory. This is done by building up a LIFO stack (this means a last in first out stack).

A program listing of the quicksort is given below:

```

10 REM      QUICKSORT
20 DIM A(100).ST(100.2)
30 RANDOMIZE (2)
40 REM      SET UP RANDOM ARRAY
50 FOR I=1 TO 100
60 A(I)=INT(RND*99999)
70 NEXT I
80 PRINT "START"
90 REM      START OF SORT
100 N=100
110 L=1
120 R=N
130 T=0
140 X=A(INT((L+R)/2))
150 I=L
160 J=R
170 IF A(I)>=X THEN 200
180 I=I+1
190 GOTO 170
200 IF A(J)<=X THEN 230
210 J=J-1
220 GOTO 200
230 IF A(I)<A(J) THEN 270
240 IF I>=J THEN 270
250 I=I+1
260 GOTO 230
270 IF I>=J THEN 320
280 H=A(I)
290 A(I)=A(J)
300 A(J)=H
310 GOTO 170
320 I=I+1
330 J=J-1
340 IF I>=R THEN 380
350 T=T+1
360 ST(T.0)=I
370 ST(T.1)=R
380 R=J
390 IF L<R THEN 140
400 IF T=0 THEN 450
410 L=ST(T.0)
420 R=ST(T.1)
430 T=T-1
440 GOTO 140
450 PRINT "END"
460 FOR I=1 TO 100
470 PRINT A(I)
480 NEXT I
490 END

```

7. SORTDEMO

Now follows a program where all the sort methods we spoke about are included. It lets you choose the number of elements to be sorted and the type of sort you want to do. The word "START" appears on the screen, the sort begins and when the sort is finished the word "END" appears. So it is possible to time the sorting. We did this with the help of this program. The results are mentioned after the program listing.

22

4.

```

5 CALL CLEAR
10 REM SORTDEMO
20 REM TISOFT
30 DIM A(400).B(400).ST(25,2)
31 DATA "1. BUBBLESORT 1", "2. BUBBLESORT 2", "3. FLAGSORT", "4. SHAKERSORT", "5. SW
APSORT"
32 DATA "6. SHELLSORT", "7. HEAPSORT", "8. QUICKSORT", "9. STOP PROGRAM"
35 RANDOMIZE (2)
36 FOR I=1 TO 9
37 READ M$(I)
38 NEXT I
40 CALL CLEAR
50 M$="*SORTDEMO*"
60 R=12
70 C=11
80 GOSUB 10000
90 M$="*****"
100 R=11
110 GOSUB 10000
120 R=13
130 GOSUB 10000
140 FOR DEL=1 TO 1000
150 NEXT DEL
160 CALL CLEAR
170 INPUT "HOW MANY NUMBERS?":G
180 IF INT(G)/>G THEN 160
190 IF G<10 THEN 160
200 IF G>500 THEN 160
210 CALL CLEAR
220 PRINT "JUST A MOMENT.."
230 PRINT "MAKING THE RANDOM ARRAY."
240 FOR I=1 TO G
250 B(I)=INT(RND*99998)+1
260 NEXT I
270 CALL CLEAR
300 REM *CHOICE OF SORT*
305 C=3
310 FOR J=1 TO 9
320 M$=M$(J)
330 R=2*J+1
335 GOSUB 10000
340 NEXT J
540 M$="MAKE YOUR CHOICE (1-9)"
550 C=5
560 R=22
570 GOSUB 10000
580 CALL KEY(O,K,STAT)
590 IF STAT=0 THEN 580
600 IF K<49 THEN 580
610 IF K>57 THEN 580
620 ON K-48 GOSUB 1000,2000,3000,4000,5000,6000,7000,8000,9000
630 PRINT "END"
631 FOR I=1 TO G
632 CALL KEY(O,M,S)
633 IF S<0 THEN 660
634 PRINT A(I)
636 NEXT I
640 FOR DEL=1 TO 500
650 NEXT DEL
660 CALL CLEAR
670 PRINT "PRESS ANY KEY FOR MENU"
680 CALL KEY(O,K,STAT)
690 IF STAT=0 THEN 680
700 CALL CLEAR
710 GOTO 300
1000 REM **BUBBLESORT 1**
1005 GOSUB 15000
1010 FOR I=2 TO N
1020 IF A(I)>A(I-1) THEN 1080
1040 X=A(I)
1050 A(I)=A(I-1)
1060 A(I-1)=X
1070 F=1
1080 NEXT I
1100 IF F=0 THEN 1130
1110 F=0
1120 GOTO 1010
1130 RETURN
2000 REM **BUBBLESORT 2**
2005 GOSUB 15000
2020 FOR I=2 TO N
2030 IF A(I)>A(I-1) THEN 2090
2050 X=A(I)
2060 A(I)=A(I-1)
2070 A(I-1)=X
2080 F=1
2090 NEXT I
2110 IF F=0 THEN 2150
2120 F=0
2130 N=N-1
2140 GOTO 2020
2150 RETURN
3000 REM **FLAGSORT**
3005 GOSUB 15000
3010 R=N
3020 F=0
3030 FOR I=1 TO R-1
3040 IF A(I)<A(I+1) THEN 3090
3050 X=A(I)
3060 A(I)=A(I+1)
3070 A(I+1)=X
3080 F=1
3090 NEXT I
3100 IF F=0 THEN 3130
3110 R=F
3120 GOTO 3020

```

```

3130 RETURN
4000 REM **SHAKERSORT**
4005 GOSUB 15000
4010 L=1
4020 R=N
4030 F=0
4040 FOR I=L TO R-1
4050 IF A(I)<A(I+1)THEN 4100
4060 X=A(I)
4070 A(I)=A(I+1)
4080 A(I+1)=X
4090 F=1
4100 NEXT I
4110 IF F=0 THEN 4260
4120 R=R-1
4130 IF R=L THEN 4260
4140 F=0
4150 FOR I=R TO L+1 STEP -1
4160 IF A(I)>A(I-1)THEN 4210
4170 X=A(I)
4180 A(I)=A(I-1)
4190 A(I-1)=X
4200 F=1
4210 NEXT I
4220 IF F=0 THEN 4260
4230 L=L+1
4240 IF L=R THEN 4260
4250 GOTO 4030
4260 RETURN
5000 REM **SWAPSORT**
5005 GOSUB 15000
5010 FOR I=1 TO N-1
5020 R=I
5030 FOR J=I+1 TO N
5040 IF A(R)<A(J)THEN 5060
5050 R=J
5060 NEXT J
5070 IF R=I THEN 5110
5080 X=A(I)
5090 A(I)=A(R)
5100 A(R)=X
5110 NEXT I
5120 RETURN
6000 REM **SHELLSORT**
6005 GOSUB 15000
6010 M=N
6020 M=INT(M/2)
6030 IF M=0 THEN 6190
6040 FOR S=1 TO M
6050 I=S
6060 J=S+M
6070 F=0
6080 IF A(I)<A(J)THEN 6130
6090 F=1
6100 X=A(I)
6110 A(I)=A(J)
6120 A(J)=X

```

```

6130 I=J
6140 J=J+M
6150 IF J<N THEN 6080
6160 IF F=1 THEN 6050
6170 NEXT S
6180 GOTO 6020
6190 RETURN
7000 REM **HEAPSORT**
7005 GOSUB 15000
7010 L=INT(N/2)+1
7020 M=N
7030 IF L=1 THEN 7070
7040 L=L-1
7050 X=A(L)
7060 GOTO 7110
7070 X=A(M)
7080 A(M)=A(L)
7090 M=M-1
7100 IF M=1 THEN 7230
7110 J=L
7120 I=J
7130 J=J*2
7140 IF J=M THEN 7180
7150 IF J>M THEN 7210
7160 IF A(J)>A(J+1)THEN 7180
7170 J=J+1
7180 IF X>A(J)THEN 7210
7190 A(I)=A(J)
7200 GOTO 7120
7210 A(I)=X
7220 GOTO 7030
7230 A(I)=X
7240 RETURN
8000 REM **QUICKSORT**
8005 GOSUB 15000
8010 L=1
8020 R=N
8030 T=0
8040 X=A((L+R)/2)
8050 I=L
8060 J=R
8070 IF A(I)>X THEN 8100
8080 I=I+1
8090 GOTO 8070
8100 IF A(J)<X THEN 8130
8110 J=J-1
8120 GOTO 8100
8130 IF A(I)>A(J)THEN 8170
8140 IF I=J THEN 8170
8150 I=I+1
8160 GOTO 8130
8170 IF I=J THEN 8220
8180 H=A(I)
8190 A(I)=A(J)
8200 A(J)=H
8210 GOTO 8070

```

```

8220 I=I+1
8230 J=J-1
8240 IF I>R THEN 8280
8250 T=T+1
8260 ST(T,0)=I
8270 ST(T,1)=R
8280 R=J
8290 IF L<R THEN 8040
8300 IF T=0 THEN 8350
8310 L=ST(T,0)
8320 R=ST(T,1)
8330 T=T-1
8340 GOTO 8040
8350 RETURN
9000 REM *END*
9010 CALL CLEAR
9020 END

```

```

10000 REM *SCREEN DISPLAY*
10010 FOR I=1 TO LEN(M$)
10020 CALL HCHAR(R,C+I-1,ASC(SEG$(M$,I,1)))
10030 NEXT I
10040 RETURN
15000 REM *RENEW ARRAY*
15010 FOR I=1 TO G
15020 A(I)=B(I)
15030 NEXT I
15040 N=G
15050 CALL CLEAR
15060 CALL SCREEN(11)
15062 CALL SOUND(100,444,2)
15064 FOR DEL=0 TO 500
15066 NEXT DEL
15070 PRINT "START"
15080 RETURN

```

5.

The sort methods were tested with this program and the time measured. The results were as follows:

	10	20	50	100	200	400
BUBBLESORT	3.5"	11.6"	1'20"	6'33"	----	----
FLAGSORT	2.7"	9.2"	59"	4'08"	----	----
SHAKERSORT	2.6"	8.4"	55"	3'33"	----	----
SWAPSORT	1.9"	6.3"	34"	2'08"	8'23"	----
SHELLSORT	3.2"	9.5"	43"	1'48"	5'26"	----
HEAPSORT	2.4"	6.3"	20"	47"	1'48"	4'06"
QUICKSORT	2.1"	5.6"	17"	36"	1'26"	3'03"

Of course, these times are approximative times. In fact the times are dependent on the length of the program. Indeed, when taking the programs apart, the sorting becomes quicker. For example, the quicksort for 400 elements takes only 2'11" in that case. From all the programs and sorts reviewed in this series it seems evident to use the fastest in your programs. It is a waste of time to use a bubblesort instead of a heapsort or quicksort. As a last remark: - there are still other types of sort.

- there are variations in the sorts themselves. The time for a Shellsort varies with the used method of sorting the groups.

BASIC Statements

BASIC Statements	PET	Apple II	TRS-80	Atari	TI 99/4	Sorcerer
General Statements						
APPEND					*	
CLS		*	*		CALL CLEAR	*
CALL address		*				
CALL CHAR					*	EN
CALL COLOR				COLOR	*	
CALL JOYSTK				STICK	*	
CALL SCREEN		HCOLOR=		SETCOLOR	*	
CALL SOUND				SOUND	*	
CLOSE	*				*	
COLOR = n	*				*	
DATA	*	*	*	*	*	*
DEF FN (name)	*	*			DEF	*
DEFINT			*			
DEFDBL			*			
DEFSNG			*			
DEFSTR			*			
DIM var(k)	*	*	*	*	*	*
DISPLAY					*	
DRAWTO		H PLOT		*		
DSP var		*				
END	*	*	*	*	*	*
EOF					*	
ERROR (mm)			*			
FOR ... TO ... STEP, NEXT	*	*	*	*	*	*
General Statements						
GOSUB linenum, RETURN	*	*	*	*	*	*
GOTO linenum	*	*	*	*	*	*
GR		*				
GRAPHICS				*		
HLIN ... AT					CALL HCHAR	
IF expr THEN linenum	*	*	*	*	*	*
IF expr THEN ... ELSE	*	*	*	*	*	*
IF expr GOSUB ... RETURN	*	*	*	*	e	*
IF expr GOTO	*	*	*	*	@	*
IN (port)		IN # expr			*	*
INPUT "msg", var	*	*	*	*	*	*
INPUT # n, var	*	RECALL		*	*	*
LET var = expr	*	*	*	*	*	*
LPRINT "msg" or LPRINT var	*	*	*	*	*	*
NEXT var						*
ON ERROR GOT linenum		ONERR	*	TRAP	e	
ON expr GOSUB, RETURN	*	*	*	*	*	*
ON expr GOTO linenum	*	*	*	*	*	*
OPEN	*				*	
OPTION BASE (x)					*	
OUT portnum, val		PR # expr	*			
PADDLE		PDL		*		
PEEK	*	*	*	*	CALL GCHAR	*
POINT			*			
POP		*		*		
POKE locn, val	*	*	*	*	(e) Call Load	*
PRINT "msg" or PRINT var	*	*	*	*	*	*
PRINT@					POSITION (e) Display At	*
PRINT #1	*	*	*	*	*	*
PRINT USING			*	*	e	
PTRIG				*		
READ var, var ...	*	*	*	*	*	*
RECALL		*				
REM	*	*	*	*	*	*
RESET (x,y)		*	*	*		
RESTORE	*	*	*	*	*	*
RESUME linenum		*	*	*	*	*
SET (x,y)		PLOT, H PLOT	*	PLOT		
SPEED = expr		*				
STOP	*	*	*	*	*	*
STORE		*			*	*
TAB	*	*	*	*	*	*
TEXT		*			*	*
UPDATE					*	
VLIN ... AT		*			CALL VCHAR	
VTAB (x)		*				
WAIT A,B,C	*	*				*

This Basic language reference chart may be of some use in transposing programs from one computer's Basic to another's. Stars indicate the existence of the command listed at the left for a particular computer. A small (e) in the 99/4A column indicates the existence of a command in Extended Basic.

System Commands

System	PET	Apple II	TRS-80	Atari	TI 99/4 Number	Sorcerer	ZX
AUTO mm, n		*	*				
BREAK mm							BR
CLEAR	CLR	*	*	*	*	*	*
CLEAR n							
CLOAD	LOAD	LOAD	*	*	OLD	*	LO
CLOAD?	VERIFY		*				
CONTINUE	CONT	CONT	CONT	CONT	*	CONT	CC
CSAVE	SAVE	SAVE	*	*	SAVE	*	SA
DELETE mm		DEL	*		*		
EDIT mm	cursor	cursor	*	cursor	cursor		cu
HOME		*					
HIMEM		*					
LIST mm-nn	*	*	*	*	*	*	*
LOMEM		*					
MAN		*					
NEW	*	*	*	*	*	*	*
RESEQUENCE mm, nn		*	*	*	*	*	*
RUN mm	*	*	*	*	*	*	*
SYSTEM	SYS	CALL - 151	*	BYE	BYE	BYE	
TROFF		NOTRACE	*		UNTRACE		
TRON		TRACE	*		TRACE		
UNBREAK					*		
(Screen Format)	40 by 24	40 by 24	64 by 16	40 by 24	32 by 24	64 by 30	32
(Character Resolution, m by n)		7 by 8	2 by 3	8 by 8	8 by 8	8 by 8	
(Total pixels)	128 full-screen	280 by 192	128 by 48	320 by 192	256 by 192	512 by 240	

String Functions

System	PET	Apple II	TRS-80	Atari	TI 99/4	Sorcerer	ZX
String Functions							
ASC (string)	*	*	*	*	*	*	*
CHR\$ (code)	*	*	*	*	*	*	*
CODE (string)							*
FRE (X\$)	*		*	*		*	*
INKEY\$	GET	GET	*		CALL KEY	*	*
LEFT\$ (string, n)	*	*	*	*	*	*	*
LEN (string)	*	*	*	*	*	*	*
MID\$ (string, p, n)	*	*	*	*	SEGS	*	*
POS (str 1, str 2, n)					*	*	*
RIGHT\$ (string, n)	*	*	*	*	*	*	*
STR\$ (expr)	*	*	*	*	*	*	*
STRING\$ (n, char)			*				*
TL\$ (string)			*				*
VAL (string)	*	*	*		*	*	*
VARPTR var				ADR			

BASIC Math and other Functions

BASIC FUNCTIONS	PET	Apple II	TRS-80	Atari	TI 99/4	Sorcerer	ZX
(Precision)	9	10	6 or 16	10	14	6	
ABS (expr)	*	*	*	*	*	*	*
ATN (expr)	*	*	*	*	*	*	*
CINT (expr)			*	*	*	*	*
CDBL (expr)			*	*	*	*	*
CLOG (expr)		*	*	*	*	*	*
CSNG (expr)		*	*	*	*	*	*
COS (expr)	*	*	*	*	*	*	*
ERL (expr)			*	*	*	*	*
ERR (expr)			*	*	*	*	*
EXP (expr)	*	*	*	*	*	*	*
FIX (expr)	*	*	*	*	*	*	*
FRF (expr)	*	*	(also MEM)	*	*	*	*
INT (expr)	*	*	*	*	*	*	*
LOG (expr)	*	*	*	*	*	*	*
MOD (expr)	*	*	*	*	*	*	*
POS (expr)	*	*	*	*	*	*	*
RANDOMIZE	RANDOM		RANDOM		*	*	*
RND (0)	*	RND	*	*	RND(1)	*	*
RND (expr)		*	*	*	*	*	*
SCRN (x, y)		*	*	*	*	*	*
SGN (expr)	*	*	*	*	*	*	*
SIN (expr)	*	*	*	*	*	*	*
SPC (expr)	*	*	*	*	*	*	*
SPC (num)		*	*	NULL	*	*	*
SQR (expr)	*	*	*	*	*	*	*
TAN (expr)	*	*	*	*	*	*	*
TI (expr)	*	*	*	*	*	*	*
USR (X)	*	*	*	*	*	*	*
AND, OR, NOT	*	*	*	*	*	*	*

TI CABLE EXTENSIONS

by D. L. Shie

My wife wanted her console on her desk over 10 feet from our 19 inch color TV. So I made an extension for the console jack to the plug of the RF modulator. I shopped at Radio Shack for 5 pin DIN plug, jack, and 5 wire shielded cable. They had the jack for \$1.50 but were out of the plugs and did not carry the cable -- they suggested multiple 2 or 3 wire cables, but I didn't want any more separate cables than necessary! I bought only their jack, and rummaged through my yard sale computer scrap. I swiped the 5 pin DIN plug and 6 feet of shielded 5 wire cable from my one good Coleco photo cell gun and 5 more feet of cable from the similar gun I had cut the DIN plug from last summer to make my monitor cable. (I may never get to my project of making light pens from those photo-cell guns!) I soldered together the two cables and the jack and taped up the evidence. My extension works fine, I can see no difference in the picture quality due to the extra long length.

I was also very interested in making cassette cables for my extra consoles I am setting up for my wife and four children. I looked closely at the various joystick extensions as they have the right kind of 9 pin connectors -- and very importantly they also have all 9 pins active so they can advertise that they will work with any of the computers -- Atari, TI, etc. I think it would work fine as a TI cassette recorder cable if one cut off the plug end and soldered on the correct three phono plugs - and from a second such cable, one could splice in the plug cut off from the other and make ones own double Atari joystick adapter with the one jack, cable, and two plugs. I bought one WICO 4 feet joystick extension with this in mind. But with the closeout TI bargains, I bought cheap TI joysticks and a TI cassette recorder with its super cable -- so the kids wouldn't have so many connections on their system. But that Wico joystick extension does work super as a cassette cable extension -- my cable was only about 2 inches short for where I wanted my recorder -- now it is over three feet too long -- but I have my recorder in its

perfect place!

Incidentally, the kids DON'T like the TI joysticks and I don't either -- as most of the rest of you told me before! And as one of you told me at a club meeting, I bought a "Power Stick" at Toys R Us and they fight over who gets to use it over the Commodore, the Atari, and even the Wico joysticks with my Atari adapter. And I and my wife prefer it also -- I may have to buy another! I see Toys R Us now carries the version for the TI also -- two sticks and one jack like TI's. It is tiny, has a very short easy throw, but seems to react much faster to control the computer -- even my two year old can work it but not the other joysticks. It did get quite a workout at the Computer show and in my UDC classes during demonstrations (its the easiest to carry around too) that the plastic ball got slightly worn at one side and it kept getting stuck in the down position for a few days, but the kids exercised that problem out of it and it again works like new.

Back to cables. I have had several hang ups with TI's power transformers -- I have both types. Either type, it is a bother to have to UNPLUG them so they don't stay hot (Prospect of a melt-down when I'm not around bothers me -- as it bothered TI to the tune of \$50 million last spring and caused a stock crash -- which led to the pullout and all these present bargains?)

Anyway, for the direct wall plug on the transformer type, I made a neat power connector: I used a heavy duty 3 wire extension cord, cut 2 feet from the single socket end and installed a double size box with 2 dual 3-prong plugs and out the other side my 2 feet long extension socket. Now on top of my P.E.Box, I have my 4 outlets with a 3 outlet converter making 6 outlets there and my other 2 foot extension goes to my transformer plug which I leave permanently upside down with the prongs sticking up and shove my outlet down on it or grab my transformer and pull my socket off upwards -- it works far better for me this way

For the other transformer with cord on

My New Keypad...Now What?
BY D. L SHIE

While shopping at Toys R Us for a gift for my Atari 2600 owning daughter, I noticed their BASIC PROGRAMMING was down to \$5. I remembered seeing that setup last September at half price of \$20 and got to wondering if the two keypads still came with it...upon checking, I discovered they had unbundled it, and were selling the "One Set Keyboard Controllers" seperately for another \$5. I bought them upon noticing that all nine of the pins seemed connected in the 9 pin plug which of course fits the TI joystick port (cassette port). Upon testing it with a joystick test program, I discovered that it does somewhat simulate the Right Joystick. The 4 key says W, the 5 key says E. And that's it for single keys. But, when I tried two keys at a time, I got a pleasant surprise (actually what I'd hoped). The 7, 8, 9, and * keys act something like the TI function keys. Hold down 7, push 4, gives NW. Hold 8, push 5, gives NE. Hold 9, push 6, gives N. And hold *, push 4, gives FIRE...but also W like 4 alone. But nothing gave any S, until... I unscrewed the case and found only 8 active wires soldered to the simple, durable key circuit board; the ninth wire was just clipped off. But enough of this wire was left, to strip it back, and solder on an extension. I noticed the Brown wire went to the top row of keys (1,2,3) which were doing nothing, I reasoned Brown went to one of the 2 prongs TI doesn't use (TI joy plug has 2 plain plastic sockets, the other 7 you can see shiny connectors). I later confirmed this with my Ohmeter testing continuity with the end of a paper clip in the plug and the the other lead on the solder connections...the Blue is the other to a socket TI doesn't use. The Grey wire (ninth one Atari cut) is the one which connects to the same socket as the empty socket on the Atari Star Raiders keypad which acts the same then as these basic keypads. I soldered the Grey's extension to the same solder point as the Brown which changes the Atari network then not at all. But on my TI, this gave me my right Joystick S directions, with the 1,2,3 acting like FCTN key: Hold 1, push 4, gives SW. Hold

2, push 5, gives SE. And hold 3, push 6, gives S. So, here I have my modified \$2.50 keypad with all the functions of the Right Joystick. I bet if I fooled around with the Blue wire the right way, I would have my Left joystick. BUT...so what? Pressing 2 keys at once is what doesn't endear our TI keyboard to many people. I tried some of the joystick games with my keypad...wow did I lose worse than usual...maybe a good handicaper for me to play my two year old with him using our best joystick? I had hoped to be able to use the thing as a numeric keypad for my TI, or as arrow controllers back in my chair, or to use it as simple input for my son. Its possible if anyone figures out the right connections for the 7 sockets TI uses. Atari could get all 128 combinations of ASCII code with 8 wires (7 bits plus 1 ground) and does get a lot with their weird basic and both controllers! Incidentally, I get nothing at all from these keypads when I try my Atari joy converter cable! At least nothing without rearranging the wires to an extent that my daughter couldn't still use them with her Atari. I may yet have to buy another set if I want to experiment further!

from 26

both sides, and the extra green check circuit breaker cord added by TI, I have installed a lamp cord on/off switch between the circuit breaker and the wall plug. Since I don't have a P.E.Box on its system (but might since I have 2 extra boxes) I now turn that computer on/off with that switch. Which seems easier for the kids to remember than the 2 other switches. But then again, it sometimes goes rotten when several of the switches are turned and only one combination will let the thing work -- almost as bad as the main TV of the house with all the connectors on the back for UHF/VHF splitter, Atari game switch, and TI RF converter -- and reaching around in the dark flipping switches this way and that for the right combination. I'm sure a lot of you know what I mean.

My family of 6 now are up to 4 TI 99's. I couldn't resist 2 more of the things at \$50 apiece. And while waiting at Toys R Us for them to be unloaded from the truck, I bought my first Disk Drive package.

Pete had given up trying to get me his 5 piece special that I had back ordered for a month when the pullout occurred, and I swore I would buy no more TI stuff, get a double sided drive from Bill, etc. for my otherwise full-blown box -- but for \$75 more than a double sided drive, getting another controller and yet a 3rd box with a single sided drive -- I weakened -- and have since also bought LOGO II, and MORE games to send for two more synthesizers for my two systems without them (I did still have receipts for four cartridges from this summer after I sold one extra synthesizer and traded one for a year's flute rental because they were of little value--and still apparently are as EVERYBODY in their right mind gets them free -- and I BOUGHT my first one for a big \$45 at WOOLCO's last gasp.) Yes, TI again has me hooked good!

Anyway, I bought the single-sided drive determined to make use of the reverse side of the double-sided disks I had bought during the October meeting group purchase special -- since we were out of the SSSD and the DSDD were still far less than SSSD in all the stores I'd then investigated.

You say I was silly buying my disks before my drive? What was worse, I bought the Editor Assembler, MSMP, and TI Writer way back in June-July and COULDN'T USE them either till I decided which drive to buy and didn't blow my money on other bargains first! And I still can't do anything with just a P-code CARD unless the club library gets the promised compiled PASCAL programs or I blow \$100's more for rest of the P-System?

Two days before our November meeting, I visited another club member and he loaned me a DSDD expensive "flippy" disk by ACCUTRACK that he had just located and found worked fine in his SS TI drive. I

went home, carefully measured, and after several tries had made a template card that matched my factory cut side and his and his factory flippy side. I took my ordinary paper punch and made my 3 punches and it worked! Then after listening to our Wabash man at the November meeting, I bought a box of his brand of SSSD disks at the meeting special and converted most of them to flippies the next weekend. I found a half round hole at the edge made with one cut of my paper punch, works just as well as the pre-cut rectangular one for write enabling, being effectively the same size. The sector reference hole is the hardest, needing two protected cuts, lifting the disk and the cover away from each other and protecting the disk with a piece of old IBM card, reaching my punch in from the center hole, and punching OUT so no burrs face the disk (as the Wabash man said his are cut at the factory). BUT my punch kept getting caught in the white lining material -- the black plastic cut fine, but I usually had to scissor off the white liner and some of that fluff still makes my cuts look ragged -- but the photocell must be positioned at the center of that large hole so when the smaller hole in the disk lets the light through, it must be away from the edge anyway -- at least the slight fluff hasn't mattered so far in over a month of use. One club member cautioned me he thought the flippies were a bad idea as then the disks turned the opposite direction in the sleeve and accumulated dirt in the liner from going one direction could then get loose when the disk was flipped and then wear my head faster. But I did this to 15 of my 22 new disks I have so far bought, all Wabash. Note NEW disks, so I risked no excessive lint build up by the disk having long spun only one way and no risk of losing any files in doing a short destructive test and initialization after the punching. In spite of the Wabash man's claim that the SSSD and DSDD disks were manufactured the same and all disks that failed the DD tests became SSSD if they could still pass the SD test, a lot of the SSSD had not even been tested at all and were probably (in the statistical sense) as good as DD or QD tested ones, but much cheaper punchable, fluff than my SSSD in which I sometimes got clean

punches right through the fluf. But they may have just been from different batches! I might just note that our Commodore friends have a much easier job of making flippies, as one told me they only need cut the side write enable hole -- so their drives must have more photocells for the multiple reference holes for possible hard sectoring, while my TI drive is a Sugart 400L which must have only one photocell and allow only soft sectoring! If I begin to have trouble with my flippies, I will only use them for backup. But so far I have had errors on 3 disks, and in all cases on the original single side, not my liberated back side, and those errors seem gone after re-initializing the bad sides -- and on those sides I vow never to keep a lone copy of anything!

With flippie disks, one big advantage of DS over SS drives is gone: Total useable disk space for a given number of disks is the same instead of double. The only remaining advantage is the total disk space available without handling the disk. I am learning TI Writer in writing this article, and am simply flipping my work-copy disk to its other side for my articles to keep them all on the same side, although the other side itself has 2/3 of its space free after the whole set of Programs copied from my master disk! I have done the same with my E/A and MSMP work disks -- which is even better as they have no useable free space on the copy side, and I must at least flip to the other side for work space. But that is a lot easier than jumping to another disk and keeping two envelopes straight!

To top it all off, I do have up to 44 sides for a total investment of \$46 -- which could have been \$36 -- about the same as if I had the same storage space on club library cassette tapes using one side only, so in a way the storage expense is mainly the difference between the disk hardware and the tape hardware for the media can be about the same price. Since disk drives are still plunging in price while older single sided drives are being heavily discounted with double sided perfected and still better technology around the corner, the only tape advantage, price, is vanishing. And with the DASD advantage over

sequential tape, I am making fewer trips to the bathroom and kitchen -- and when I do get up, I power my computer down more, as its easier to power back up and retrieve my work now!

PROGRAM LIBRARY
by
Larry Hughes

Many thanks to the following members who have generously donated programs to the Club library during the past month: Christopher Bobbitt, Jim Horn, Dave Ramsey, Joel Thrasher, Doug Smith, and Heather Smith.

Thanks to these members, plus previous donors, we have another full disk/tape! Tape B30 will be available for pickup at the next regularly scheduled pickup meeting, March 8th. Disk B30 may be ordered at the Feb. 9th meeting.

Also, more programs have been added to fill out the half full disk/tape B25 and it may be obtained at the same meetings.

And an especially great big THANK YOU to new member PHIL SIMERLY who has donated over 500 programs to the Club library!!! (However, probably 80% of them are already in our library).

Needless to say, it will take me quite some time to run each program, eliminate the duplicates, rename the ones that require Extended BASIC, and consolidate them onto several disks. I will try to come up with one or two new disks/tapes each month. Thanks again to Phil for his great generosity!

COMPUTER, KIDS AND CRIME

The FBI Criminal Investigation Division has agreed to address the User's Group on the subject of computer crime. This presentation will be given at the February meeting on Thursday the 9th. Invite your friends and come early.

COMPU-SOURCE CORNER

By Rick Stickle

TI0121

70220,242

This should be one of the shorter columns this month due to lack of time and news. During the last month I have changed from a Washingtonian to a Marylander. I still welcome all questions you have about getting on line with other computers, I can be reached at the above account numbers on the respective services or my new phone number is (301) 725-6979.

A hint for getting on line, most people who call seem to have trouble setting up their TE II and modem. I suggest that you start using the default settings for your TE II and full duplex on the modem. These settings will work on 90% of the networks you access. You must also remember that the 99/4 will not normally supply line feeds, if a system asks you if you need line feeds answer yes then ask for 1. A sure sign you should have asked for a line feed is when the line prints over the previous one so you end up receiving only one line at a time from the host computer.

I can't close with out mentioning the fact that TI users in the area recieved a Christmas gift! If you dial (301) 434-0117 you will find that the area now has its own TIBBS. This system is running 24 hours a day on a TI 99/4a and has been placed in use by SYSOP Phil using the same program that is used in Atlanta, GA. Most of the crew from MAGUS and the old SMART board can be found there and at the present time it is not hard to access, I'm sure this might change as more TI users find the board.

One last thing worth mentioning, I found a book called Free Software for your TI-99/4a it is written by David and Dorthy Heller and has a real good section on using your TI on line, not to mention a write-up on the users group here in Washington. The price on this little gem is \$8.95 at your local book store. It also includes a listing of public access message systems (thats bulletin boards to us common folk)

Starting with this issue your newsletter will have a swap corner. This is open to any club member who has an item to sell or trade. Just mail your item or items to R. D. Sturgell, 801 N. Pitt St., #303, Alexandria VA., 22314. No comercial ads will be accepted. So do you have any used or unwanted items. Why not trade or sell them. Let's fill a page next month. Just tells us what you have to trade or sell, your name, and your telephone number.

INFORMATION RESOURCES

This list of members consists of individuals who have volunteered to be pioneers in developing our "infrastructure" as it might have been called in the sixties. Networking is more popular today. Our members skills are a primary resource of our clubs. If you see categories of interest not covered, please call Dave Hill, 941-6876. Resource volunteers should also call Dave with your replacement's name and telephone number. More than one resource volunteer is OK, because of driving time and long distance calling.

PLEASE BE BRIEF AND EARLY

Some people accept calls at any time. If that is true, they will tell you so. Nine fifteen pm is about the latest you should call a volunteer otherwise.

SUGGESTIONS ARE CHEAPER THAN WORK

Volunteers enjoy what they are doing, but, like you, have more ideas for new projects than they have people to do them. Resource volunteers have only agreed to put you in contact with others of similar interest, essentially by giving you the names and phone numbers of persons who have called with similar interests. Your help will be welcome along with your suggestions. If you choose to get in the thick of things and become a telephone Resource volunteer, you can personally gain a great deal.

PAUL PEYSER ASSEMBLY LANGUAGE
649-9870

DAVE HILL RESOURCE EDITOR
RESOURCE

941-6876 MULTIPLAN
TI WRITER

KEN CHANDLER NOVICES
752-9774 SOFTWARE REVIEWS

RICK STICKLE EDUCATION, CAI,
301-725-6979 LOGO, MODEMS,
TELECOMMUNICATIONS

GARRY CRADDOCK VIDEO DISK
703-494-4952 TECHNOLOGY

DICK STURGELL YOUTH GROUP
NOVICE INFORMATION

CHRIS GOODMAN CP/M
262-5570

M. SWINNEN COMPACT COMPUTER 40
459-5458 CALCULATORS

PETE EDDY GROUP PURCHASE
631-1744 COORDINATOR
IMMEDIATE PAST VICE
PRESIDENT

ADVERTISERS SEE OUR ADVERTISERS
FOR ADVICE ON
EQUIPMENT
SOFTWARE

MIKE LAMBERT PASCAL
920-3955 "C"
FORTH, GPL, PILOT

RON BURKTON HARDWARE HACKERS
703-368-7187

DAVE ROTHMAN HACKING SINCLAIR
654-5584 COMPUTERS

BILL WHITMORE PUBLISHING COMMITTEE
DICK STURGELL

PHIL SIMERLY COMPUTER BULLETIN
301-434-0117 BOARD

301-645-1432 MEETING INFORMATION
RECORDING

ALLEN MINTON HOUSHOLD PROCESS
493-4502 CONTROL, MATH,
MUSIC
PRESIDENT MONTGOMERY
COUNTY TI USERS GRP

GEORGE SMYTH ADVENTURE, ROLE
703-533-8710 GAMES (ETC)

Doug Smith Cartridge Software
301-645-1432 Review Exchange

FOR SALE
TEXAS INSTRUMENTS

Cassett Cables (single)	\$20
TI Cassette Recorder	\$55
with cable	\$70
GE Cassette Recorder	\$30
with cable	\$55
Recorder dust cover	\$ 4
WICO COMMAND joy sticks	\$40
TI Joy sticks	\$20
TI99/4A Computer	\$150
Command Modules:	
Football	\$15
Munchman	\$15
Extended Basic	\$80
Teach Yourself X-Basic	\$20
Teach Yourself Basic	\$25

All in excellent condition. Very little use. Call 642-8632 after 5 PM. if no answer call 241-8942 Annandale, Va.

CorComp - RS232 CARD

Latest news is that CorComp wired the output of the RS232 card somewhat different than TI had. This will lead to mass confusion when users try to interface the RS232 with other peripherals. If you have a modem it will work OK if it is not a TI or TI configured modem. Any printer you have will not work if you use a ribbon cable or a prewired pin-to-pin cable. I hope that this rumor is untrue, if it is how could they goof like this?



HOME BUDGET

Did Holiday expenses get you again this year? Using HOME BUDGET you can help yourself to solve this annual headache by planning ahead. This is what budgeting is all about. With the help of your TI-99/4A, 32K memory expansion, disk system (Extended Basic required) and printer you can EASILY plan ahead for Holiday and other financial goals. HOME BUDGET will sort your checks by check number or account number (you assign titles and numbers for up to 50 different categories) Other printouts include: (1) budget table with monthly budget and expense, annual budget and expense and over/short information, (2) Table showing average and annualized expenses and (3) a table showing the source of your deposits and the balance in your checking account. This program was written by an accountant and is guaranteed to work. While it won't stop you from overspending it will let you know on what course your financial ship is headed. As with all things in life the key to success is PLANNING. Financial planning, the key to success for business, is now available to you at home with your TI-99/4A and HOME BUDGET. Send \$19.95 (VA residents add 4% sales tax) to: DCH Software 7010 Catlett St. Springfield, VA 22151.

HOME BUDGET - \$19.95

(Name)

(St. Address)

(City, State & Zip)

Mail to : DCH Software 7010 Catlett St.
Springfield, VA 22151

By Bill Tiep

The subroutine below has appeared several places, but I thought some of our newer members might enjoy trying it out.

In order to display a message in TI BASIC at any Row/Column location on the screen, you just assign the string to MSG\$, the Row number to ROW and the Column number to COL, then GOSUB 9000.

```

100 REM DISPLAY A MESSAGE
110 REM
120 CALL CLEAR
130 MSG$="MESSAGE"
140 ROW=12
150 COL=13
160 GOSUB 9000
170 REM
180 REM MORE PROGRAM...
190 REM
200 END
9000 FOR I=1 TO LEN(MSG$)
9010 V=ASC(SEG$(MSG$,I,1))
9020 CALL HCHAR(ROW,COL+I-1,V)
9030 NEXT I
9040 RETURN
    
```

Here's how it works. LEN(MSG\$) gives the length of "MESSAGE" which is 7. This is the number of iterations of the FOR-NEXT loop.

SEG\$(MSG\$,I,1) pulls out each character in the string in succession. That character's ASCII value is assigned to V by the ASC function.

Then the CALL HCHAR statement puts the character on the screen at the specified row and increments the column by I.

When using the subroutine remember that HCHAR uses a 32 column screen!

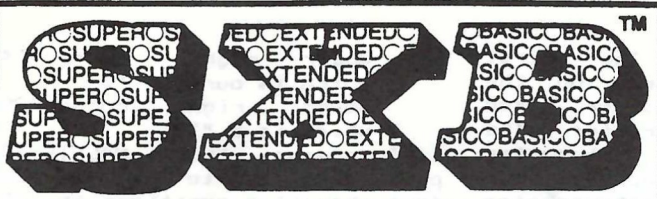
FROM NEW
HORIZONS

PROGRAM DEMONSTRATION AT 3/22
MONTGOMERY COUNTY MEETING

Larry Hughes, President of QUALITY 99 SOFTWARE, will demonstrate five of his useful disk programs at the March 22nd meeting of the Montgomery County Users Group.

The programs to be demonstrated are: QUICK-COPYer (tm) --- backs up a disk in four passes or less; QUICK-Cataloger (tm) --- a resident disk catalog program; EZ-LOADER (tm) --- loads and runs any program at the touch of a single key; DISK LABELER (tm) --- prints a disk catalog on a mailing label which may be affixed to the disk envelope; THE LIBRARIAN (tm) --- a multi-disk catalog program which produces an alphabetized listing of up to 50 disks.

Come to the meeting, see these programs in action, ask questions.



A POWERFUL GROUP OF MORE THAN 100 TMS9900 ASSEMBLY LANGUAGE SUBROUTINES WHICH, IN CONJUNCTION WITH EXTENDED BASIC AND MEMORY EXPANSION, SUBSTANTIALLY EXPAND YOUR PROGRAMMING CAPABILITIES ON THE TI-99/4A HOME COMPUTER:

- **DATA BASE Subroutines** include SORT with multiple ascending and/or descending keys.
- **STRING ARRAY Subroutines** include encryption, transcription, copy and viewing.
- **STRING Subroutines** include data compaction and conversion.
- **INTEGER Subroutines** allow storage of 4 times as many numbers in a numeric array.
- **V.D.P. Subroutines** include screen key entry, reading, listing & drawing + lightening-fast character definition/retrieval + FG/BG & sprite color changes.
- **Plus many other useful subroutines** including parameter passing between programs.
- **Easy to use** — knowledge of assembly language not required.
- **Disk or Tape Software** with loose-leaf instruction book only \$99.95 postpaid (plus 4% sales tax in VA).

J & K H SOFTWARE — Dept. W
2820 South Abingdon Street
Arlington, Virginia 22206
☎ (703) 820-4131

Also Available

VIDEO TITLES I (Disk or Tape)	\$29.95
VIDEO TITLES II (Disk, XB)	\$49.95
VIDEO TITLES III (Disk or Tape)	\$24.95

SMART PROGRAMMING GUIDE™ FOR SPRITES 5⁹⁵

Plus Shipping

This guide will show you some of our professional programming secrets on how to: Use CALL PEEK • Get sprites to pick up objects, eat dots and lay down a trail. • Shoot sprites without missing a coincidence. • Make one sprite chase another. • Easily convert sprite rows and columns into graphic rows and columns and visa versa. • Generate moving sprite patterns. • Use 3 different CALL KEY or CALL JOYST examples for moving sprites. • Write a GENERAL BAR GRAPHING program (to one pixel accuracy) that shows you sprites aren't just for games.

Full of fast running and Byte saving examples that you can use in your existing programs or combine together to write your own programs. Each example program is fully documented in a step by step method that is easy to understand. A TI 99/4 or 99/4A computer and the extended basic command module are required.

Sorry, no C.O.D.'s or credit card orders. Foreign orders payable in U.S. currency. CA. residents add applicable sales tax. Shipping and handling U.S., Canada and Mexico 1.50. All other countries 3.50.

15 day money back guarantee.



MILLERS GRAPHICS

1475 W. CYPRESS AVE., E-1 • SAN DIMAS, CA 91773
(714) 599-1431

FREE CATALOG AVAILABLE

GO WITH THE FLOW

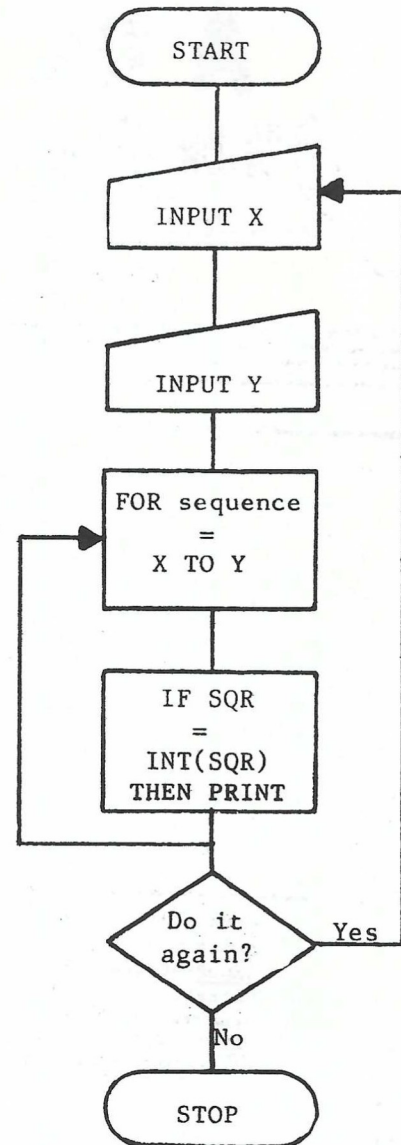
In real life we do not sit down at the keyboard and write a program straight into the computer. Only the foolish or overenthusiastic would attempt such a thing. Instead we outline in some way what the program is supposed to accomplish, and continue on from there.

Flowcharting is one of the accepted and professional methods of outlining the structure of a program. It clarifies the logic that you have in mind as you proceed from the idea of what the program is supposed to do, to the finished product in proper BASIC, or another language. Flowcharting makes program writing easier, more coherent, and therefore more likely to work right after it has been converted into proper code.

Now let us make a flow chart for a program that will print out all the numbers whose square root is a whole number. The program will do this between any two whole numbers that we choose and upon completion ask us if we want to do any others.

You could probably look at that flow chart and write a program in BASIC that would work the first time. In actuality very few people think in terms of flow charts with boxes and lines showing the possible paths. Most people think in terms of sentences or phrases. Using this for program outlining is an acceptable alternative to flow charting. It is called pseudocode.

Remember that earlier paragraph that began with..."Now let us...." I used that paragraph to say what I wanted my program to do. With a little change it can completely replace my flow chart as a means to develop an outline of a projected program. Here is the pseudocode example.



INPUT two whole numbers X and Y
 FOR a sequence of numbers starting at X and continuing till Y
 Check to see IF the SQR of each number in the sequence is an INTEGER; THEN print the number and the SQR of the number.
 INPUT whether to start again or not with new numbers. IF the answer is yes go back to where the numbers were first input.
 STOP

++++++LATE NEWS++++++

If you have a P-Code Card and sent in a card for a free spreadsheet in Pascal ---- forget - it.
 However you may soon be receiving a Microsoft Multiplan.

Everything we do is
A OK
 at
A OK Printers, Inc.
 9425 Mathy Drive
 Fairfax, Virginia 22031
(off Pickett Rd behind Pickett Shopping Center)
273-6215

CREATIVE DESIGN • TYPESETTING • XEROX • BUSINESS STATIONERY
 RAISED PRINTING • ANNOUNCEMENTS • NCR FORMS • FLYERS • LABELS
 BOOKLETS • WEDDING INVITATIONS • FREE PICK UP & DELIVERY

NOTICE TO MEMBERS OF THE WASHINGTON DC AREA TI HOME COMPUTER GROUP. Please check your mail label for the renewal date of your Membership. If your date is JAN, FEB or MAR 1984 the date will be circled in RED. We are now trying to group the expiration dates so that the mail list will be easier to maintain. Your dues will be prorated. Thank you for your cooperation. If your date is circled in red please remit JAN \$12.00, or FEB \$11.00, to extend your membership until JAN 1985, for the labels marked MAR please remit \$16.00 to renew until JULY 1985.

ROBINS Inc.

THE SMALL COMPUTER SUPPLIES STORE

LABELS 3 1/2 x 15/16 (5000) \$16.00	PRINT WHEELS Diablo Plastic 6 40 Diablo Metal 41 50 & 49 95 Datiko 620 11 40 QUME 6 40/7 90 Smith-Cor TP 1 4 90 Olivetti 15 50
PAPER (15, 18 & 20 lb.) 9 1/2 X 11	OLYMPIA 26.50
RIBBONS Radio Shack Daisy Printer II 4.75 each Line Printer III & IV 6.50 each Line Printer VI & VIII 6.50 each Diablo 620 M/S 7.45 each Hytype II M. S. 46 50 doz. NEC 5500, 7700 M. S. 63 00 doz. Cloth 69 90 doz. 3500, 8023 11 90 each C. ITOH Prowriter 11 90 each	DISKETTES 5 1/4 ss/DD 26.50 VERBATIM 33.00 MAXELL 33.00
QUME M/S I & Cloth 33.50 doz. M. S. IV 54.00 doz. IDS 560 5.75 each EPSON MX 70 80 6.00 ea MX 100 12.00 ea	RIBBONS FOR OLYMPIA \$4.00 ea 10/37.95 CORR TAPE OLYMPIA Box of 10 \$16.95
LOAD YOUR OWN! Refills for Diablo, QUME RICOH NEC 1.50 EPSON MX 70, 80 3.50 MX 100 8.99	560-5900 8304 Hilltop Road Fairfax, Virginia 22031 (Merrifield Area)

Hours: Monday thru Friday 9-5, Saturday 10-2

MEMBERSHIPS: Memberships are available to the general public. Cost for dues depend on the group you join. Membership information for each group is: WASHINGTON D. C. AREA TI USER'S GROUP--Contact Frank Jordan at (301) 899-3707 or mail your dues (\$12) to Frank at 4100 Canterbury Way, Temple Hills, Md. 20748; TI BALTIMORE USER'S GROUP (TIBUG)--Send \$15 to Membership Chairman TIBUG PO BOX 3, Perry Hall, MD. 21128; HAGERSTOWN-WILLIAMSPORT TI USER'S GROUP--Write Sam Williams at PO BOX 376, Williamsport, MD 21795 or call (301) 223-8014; MONTGOMERY COUNTY TI USER'S GROUP--Call Allen Minton at (301) 949-4085.

TI NEWS
 BOX 267
 LEESBURG, VA
 22075

TIME SENSITIVE MATERIAL

BULK RATE
 U. S. POSTAGE
 PAID
 LEESBURG, VA.
 PERMIT No. 68

MEMBERSHIP DUES \$12.00
 WASHINGTON D. C. AREA
 TI 99/4 USER'S GROUP
 MAIL TO: F. Jordan
 4100 Canterbury Way
 Temple Hills, MD 20748
 Z TI Membership

Dudley Perry
 423-2895

NAME _____
 ADDRESS _____
 CITY _____ STATE _____ ZIP CODE _____
 TELEPHONE _____ INTEREST _____

Hagerstown - Fud
 Balt 1st wed ead mo
 Bowie - 2nd wed Sev. Park

Duane Shie
 703 420-9693