

DALLAS TI HOME COMPUTER GROUP
SEPTEMBER, 1985

Meetings: 7:00pm, 3rd Friday each month
At Northlake College, Room B-206, Irving, Texas

>>> NEXT MEETING: Friday, 20 September, 1985 <<<

President: Richard Roberts (SOURCE: T13352; STARTEXT: 8762)
Vice-President: Dan Johnson
Secretary: Louis Guion (STARTEXT: 77536)
Treasurer: Earl Bullock
Editor: Robert Lee Hoffpauer

This newsletter is the official publication of the DALLAS TI HOME COMPUTER GROUP, a non-profit organization serving member/users of the Texas Instruments 99/4A HOME COMPUTER. For more information you are invited to attend our next meeting or send a SASE to: DALLAS TI HOME COMPUTER GROUP, ~~1000 Northlake College~~, Irving, Texas 75061.
.....

PRESIDENT'S SCREENDUMP: Richard Roberts

Things sure do change fast. Just a month ago, we still had vacations and hot weather on our minds, and now school has started and it seems more like fall already. Hopefully, everyone is back into their familiar routines again, and will be paying more attention to their trusted 99/4A. There is certainly a lot that has happened in the 99'er world in the past few months, and you will want to catch up on all the latest, if you've not been at the User Group meeting lately.

Last month saw the beginning of Robert Hoffpauer's series on BASIC programming, and he will deliver the second part this month. Also, we will demonstrate PAINT 'N' PRINT, one of the newest graphics programs on the market.

The PNP program comes to us courtesy of Navarone, who has recently initiated a program of allowing User Groups to demonstrate their latest software at group meetings, in exchange for an honest evaluation of that program. This will allow group members to see first hand what the software is capable of, before going out and plunking down big bucks sight unseen.

This promises to be a big benefit for us, as Navarone has almost a dozen new programs under development, and due to be released within the next year. And, the program is ours to keep. The Executive Committee has voted to offer the program for the main raffle prize, after the demonstration is complete.

Speaking of the raffle, congratulations to Guy Hunter who won the Adventure Editor at the August meeting. Remember, the raffle is open to anyone in attendance, member or guest. Each entry is \$1.00, enclosed in an envelope with your name on the outside. You may enter as many times as you wish. The raffle box and the envelopes are available at the sign-in table. Listed below are the three items available at the September meeting, from which the winner may choose one item.

- > PAINT 'N' PRINT (can be used with console only)
- > MULTIPLAN spreadsheet program (requires full system)
- > Return to Pirate's Island (can be used with console only)

We have made a change in the meeting date of the Executive Board. The meeting will now be held the first Saturday following the User Group meeting. The time is still 1:00 PM, at Louis Guion's house. The next meeting will be September 28. Please mark the date on your calendar.

I'm looking for a really good turnout this month, so come on out and enjoy the fun.

SECRETARY'S MINUTES: Louis Guion

August's meeting of the Dallas TI Home Computer Group was held on August 16, 1985. The meeting was called to order at 7:17 P.M. by Secretary Louis Guion, acting in behalf of President Richard Roberts who was unable to attend the meeting due to an illness. Vice-President Dan Johnson, back with us at last, said he had a "stubby" pencil and could act as Treasurer while Jackie Guion recorded the goings-on as stand-in Secretary. This crew got things rolling, finally! The minutes from the July meeting were read and were accepted.

One of the three visitors present joined our users group. He is Greg Justice of Garland. The other visitors were Jim Leshner and Doug Garretson, both of Dallas.

NEW BUSINESS: An announcement from Miller Graphics concerning a forthcoming product to enable the downloading of cartridges was read to the group. A descriptive letter concerning GTE's PC-PURSUIT was passed around for reading as was the copy of a session on PC-PURSUIT's BBS. A copy of the COMPUTER SHOPPER, printed in Florida, was sent down the rows for inspection.

Robert Hoffpauer presented the 1st in his series of "Programming in Basic" lectures. He discussed and entered, with John Guion's help at the keyboard, a BUBBLE SORT program. Robert wrote on the board so that the audience could copy the code while John entered the program into the club's system for "instant demonstration".

To give a break between program segments, and to cover Louis' late return from a Coke/cola run, Bob Viering presented a report on current happenings at the club's TIBBS bulletin board, the 99'er Connection. We introduced those of our guests who had completed the questionnaire.

An "Open Forum" was held. A few of the highlights were the announcement of where some disk controllers, disk drives, and other hardware and software had been located by three or four members and the offering for sale of several items owned by members. A cassette/console interface cable and two cassette tapes covering start-up use of the TI-99/4A were given away to deserving (?) members.

Bob Viering presented the second program of the evening, covering GRAPHX. Bob did a great job on only a scant's notice. He had to take over when President Roberts was unable to attend. Some samples of GRAPHX output were shown. They were a name badge and copies of the "clipboard" art that comes with GRAPHX. Never content, Bob even volunteered to send in another order for GRAPHX — and was taken up on his offer by over five people before the meeting adjourned.

The "dollar" raffle was won by Guy Hunter who chose the Adventure Editor as his prize.

The meeting adjourned to Crystal's Pizza at 9:00 P.M.

Setting Down to Basics: Robert Lee Hoffpauer

The simplest program structure is the sequence — one instruction after another, after another, and so on. The computer will perform the instructions in sequence from beginning to end without deviation. However, there are times when it would be convenient to deviate from the sequence. BASIC has very few control structures to accomplish these deviations, but what is provided is enough. We briefly discussed subroutines last month — subroutines allow us to write a segment of instructions as a separate module, call that module from somewhere else in the program, then when the module has been performed, we are returned back to where we were. This is about as fancy as it gets, folks, so enjoy it.

There are, however, some fundamental program control structures that come into play — and much more frequently than subroutines. This month I want to discuss loops and looping constructs. Loops are simply a way of performing a segment of instructions repeatedly. TI BASIC, and most other BASICs only provide one (1) built in looping construct — the FOR-TO-NEXT control structure. Example 1 is a sample of a FOR loop that executes it's code eight times — it reads data from a data statement into an element of an array — a different element each time through the loop. To repeatedly perform some set of statements is called iteration.

Loops can be nested like a set of little boxes that fit inside one another. The only criteria is that an inner loop must terminate before it's outer loop terminates. Example 2 is a sample of loops nested to two levels. The subject of FOR loops is covered in your User's Reference Guide starting on page 11-53.

entities and to the overall program design, as well.

What are data entities? Creatures from outer space? No. Data entities are the creatures of the inner space of the computer's memory. The computer cannot guess what you want it to do, or what data is important, or how to store that data for easy recall. Your program must provide certain structures for temporary storage of data during the time the program is running — these are usually called variables because the structures you provide can hold values that vary from moment to moment, as long as they can 'fit' into the structure provided. You can't fit somebody's last name into an area set aside for numeric values, but an area for numeric values can hold the values 13, 54, -45535, or zero. The actual value can vary, hence variables. BASIC language programs are easy for beginning programmers in this area of data structures because there are essentially only two structures provided. Other languages provide a variety of very sophisticated structures, but almost anything you may want to do can be achieved with the two provided in BASIC.

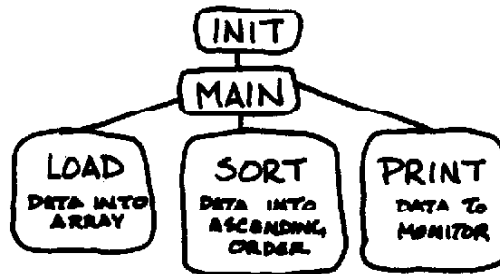
These two BASIC language data structures are the variable and the array. We mentioned the variable, above. The array is simply a collection of related variables of a similar type; i.e. all numeric, or all alphabetic. As I'm sure you already know, when you set up a variable you give it a name, such as LASTNAME, FIRSTNAME, or AGE. Likewise, when you set up an array you give a name to the whole array. You will refer to the individual elements in the array using subscripts. For instance, if you had an array of names, you might refer to the third name as NAME(3). So, the next step is to decide what data structures your program will need and give them meaningful names. For the rules to use in giving names to variables and arrays, see page II-11 of the User's Reference Manual supplied with your computer. In fact, that manual, the Extended BASIC Manual, and the Disk Memory System manual (if you have a disk drive) are primary references for programming our 99/4A computers in BASIC and XBASIC.

When you think you have the data structures listed and named, you are ready to think about the overall structure of the program. BASIC is roundly condemned by many advocates of structured programming because it does not enforce structure, and encourages what is called spaghetti code with its frequent GOTO statements. While this is true because BASIC was developed before the advent of structured programming, the programmer can give his BASIC program a measure of structure, and will benefit by doing so. The two primary benefits of structure in programming are, first, the readability of the listing, and, second, the ease with which the program can be de-bugged and modified. I can guarantee that you will have bugs in your program, and it is a safe bet that you will want to modify it. A third benefit is the ability to write standard routines to accomplish frequently used tasks, so that the same routine can be used in several programs without having to reinvent the wheel each time. A good example of this would be a routine to sort a set of data into proper order.

So how do we structure a program, already?! By the tactic of divide and conquer. Using your pencil and paper again, draw a sketch of the various 'modules' you will need for your program, similar to the illustration further on in this article. The INIT module will be short, appearing at the beginning of your program, and will initialize anything that must be set to an initial value, as well as setting the dimensions of your arrays. The MAIN module will follow immediately, and this module is the executive whose function is to run the rest of the program by calling all of the subsidiary modules as they are needed. BASIC provides the ability to create all the subsidiary modules as subroutines (see: pages II-113 thru II-117 in the User's Reference), while XBASIC provides a more sophisticated subprogram capability. Once you have decided on the basic structure of the program, you are finally ready to begin writing the code for the various modules. If you have applied the principles and procedures that I have outlined this month and last, you will find that even when you have problems, they can be isolated easily in one module, and corrected easily by modifying the one module. If you have been trying to develop a program in a spaghetti code fashion in the past, you will notice the difference: if not, just take my word for it. Happy trails!

PROGRAM: SORTDEMO

NOTICE THERE ARE NO DIRECT PATHS BETWEEN MODULES, YOU ENTER FROM THE 'MAIN' MODULE AND RETURN BACK TO 'MAIN'. THUS,



'PRINT' COULD BE CALLED BEFORE 'SORT' TO SHOW THE ORIGINAL ORDER, THEN CALLED AGAIN AFTER 'SORT' TO SHOW THE FINAL SORTED ORDER. RLA

any that would like to take part in guiding the direction of the club. Our meeting space is small, so let Louis or myself know in advance, if you plan to attend.

We should have a good meeting this month, so make your plans now to attend on August 16th. We will try to start the meeting at 7:00 p.m., sharp. I'll be looking to see everyone there.

SECRETARY'S MINUTES: LOUIS GUION

The July meeting of the Dallas TI Home Computer Group was held on Friday, July 19, 1985. The meeting was called to order at 7:15 P.M. by club President Richard Roberts. The minutes from the June meeting were read by Louis Guion and were accepted by the membership.

Visitors, having been welcomed at the special table set up in the back of the meeting room, were: Jerry Johnson of Irving; Robert Maynard of Dallas (one of our newsletter subscribers, but not a member!); Cindy Kelly of Irving; Warren Brinberg of Richardson; and Rick Button, also of Richardson. Cindy, Warren, and Rick joined our club before the meeting was over. Old members Stephen Schaub and John Smith renewed their memberships, and Jimmie Kennedy was added to the existing single membership of her husband, Dr. L. H. Kennedy. Robert Maynard renewed his newsletter subscription.

NEW BUSINESS: There being no old business, the club heard a report on the release of Vice President Dan Johnson from the hospital. The members felt that Dan might enjoy a complimentary subscription to Startext and voted to request the treasurer initiate a membership for Dan. President Roberts presented a short discussion on the virtues of Startext for those not already familiar with the service.

PROGRAM: Keith Althar, Charles Howard, and Charlice Althar supplied a demo program, an insurance application, and a mailing list set-up for demonstration during the Multiplan applications session. Richard Roberts acted as host, moderator, and part-time demonstrator. The authors of the demonstrated works each presented special highlights of their creations. Charlice won the "Disk Jockey" diskette and label holder awarded for "the most unique use" of Multiplan.

RAFFLE: Iagene Osborne won the "dollar raffle" and chose "Teach Yourself Basic" as her prize. After three meeting's anticipation, the drawing was finally held for the club's old P-800 drive. The winner? David Martin, our Hospitality Chairman!

The meeting adjourned to Crystal's Pizza and small talk at 10:00 P.M.

EGG SALE: 99/4A console, joystick, and four modules (Invaders, Munchaus, Home Financial Decisions, Personal Record Keeping). Asking what was paid -- \$95.00.

Jerry Johnson
256-1142

Getting Down To Basics: Robert Lee Hoffbauer

Last month we covered the steps to follow in deciding what it was we wanted to do: (1) write out a description, (2) list the inputs and the outputs we will need, (3) make sketches of all outputs (as well as input prompt screens), and (4) layout the record forests of any data files we anticipate using. When we had done this, we set it aside, for later review. If, on later inspection, we were satisfied with our work so far, we said we were ready to begin designing the program -- not coding it yet, but designing it.

Most amateur programmers (myself included) and many professionals are in too great a hurry to get to the coding stage, but study after study shows that a better program with fewer errors (bugs) can be generated in a shorter amount of time by being methodical and planning. There is an old saying that applies, "Plan your work, and work your plan."

In the world of computer programming there is a concept called 'structured programming,' which is essentially what we have been talking about so far. Now to carry the concept one or two steps further, we will apply structure to data