HOOSIER USERS GROUP HOOSIER USERS GROUP HOOSIER USERS GROUP HOOSIER USERS GROUP HOOSIER USERS GROUP
USERS GROUP HOOSIER USERS GROUP HOOSIER USERS GROUP HOOSIER USERS GROUP HOOSIER USERS GROUP HOOSIER
GROUP HOOSIER USERS
HOOSIER USERS GROUP
USERS GROUP HOOSIER
GROUP HOOSIER USERS
HOOSIER USERS GROUP

THE HUGgers
HOOSIER USERS GROUP
People Helping People

GROUP HOOSIER USERS
HOOSIER USERS GROUP
USERS GROUP HOOSIER
GROUP HOOSIER USERS
HOOSIER USERS GROUP
USERS GROUP HOOSIER USERS GROUP HOOSIER USERS GROUP HOOSIER USERS GROUP HOOSIER USERS GROUP HOOSIER
GROUP HOOSIER USERS GROUP HOOSIER USERS GROUP HOOSIER USERS GROUP HOOSIER USERS GROUP HOOSIER USERS

DECEMBER 1992 | The HUGgers Newsletter | Volume 11, Number 12

MONTHLY MEETING LOCATION
LITTLE HOUSE NEXT TO
ST. ANN'S SCHOOL
2839 S. McCLURE
INDIANAPOLIS, IN

MEETING STARTS
AT 2:00 P.M.
DECEMBER 20 1992

OFFICERS' CORNER

It seems that none of the officers have either had anything to say (or got around to say it) this month. So I'm typing this up as Bob is pasting things up.

I don't know what the rest of page one is going to look like. Bob's threatening to fill the rest of the page with quips of some kind.

Bill Lucid reports that the new BBS software still hasn't shipped in its working version. The author has been having various hardware failures in addition to being a full-time student. He hopes to have it out by the first of the year.

A big welcome back to Bob Mikels. His renewal brings us up to 30 members! (The big "three zero" as they say.)

Some of you may know that there is a difference of opinion between us and the copier guy concerning our maintence contract. I expected it would either have been or close to being resolved by now, but it hasn't. Right now we're waiting for Cyclone to send us more information.

That's about it for now. See you all at the meeting.

-GBL

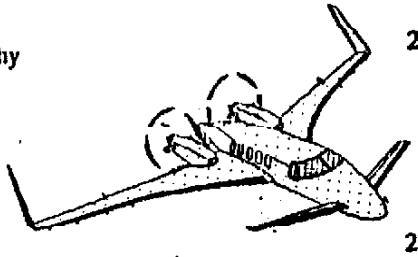
Our BBS and You

The 30 Commandments of BBSing

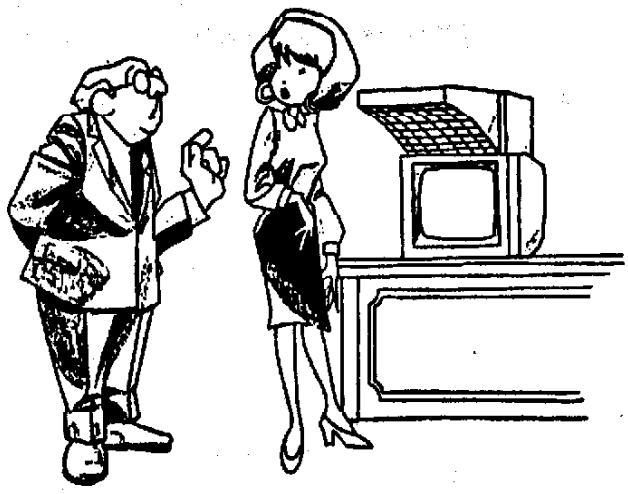
(Author unknown) (Downloaded from Technoids Anonymous BBS, Phoenix, AZ.)

Some very intelligent SysOp gave some great thought to how a BBS should be operated and came up with these 30 BBS Commandments.

1. Thou shalt love thy BBS with all thy heart and all thy bytes.
2. Thou shalt remember thy name and thy password.
3. Thou shalt speaketh to the SysOp thy real name.
4. Thou shalt not POST IN ALL CAPS!
5. Thou shalt Honor thy SysOp.
6. Thou shalt not covet thy neighbor's password, nor thy neighbor's real name, computer, software, nor any other thing of thy neighbor's.
7. Thou shalt only call a BBS two times a day.
8. Thou shalt not post messages that are stupid, worthless, or have no meaning, lest thee be deemed a fop.
9. Thou shalt use the English language properly so that thee may be deemed wise.
10. Thou shalt spell thy words correctly. Amen.
11. Thou shalt delete thine own mail.
12. Thou shalt delete thine ancient mail.
13. Thou shalt help other users so that thee may be deemed a friend.
14. Thou shalt not post anonymously when offering criticism.
15. Thou shalt keep thy foul language to thyself lest thee be forever banished.
16. Thou shalt not occupy thy BBS with thine arguments, for Verily, I say unto thee that thou shalt maketh thyself the charlatan.



17. Woe be unto the user who attempth to crash thy BBS, for such shall be cast out from the sanctuary of thy hobby and must repent by doing 40 mega-days and 40 mega-nights of penance in voice-only communications.
18. Thou shalt not post PRIVATE messages, lest thou truely has something to hide.
19. Thou shalt not post other user's real names.
20. Thou shalt not post messages after imbibing excessively of ale.
21. Thou shalt confine thy messages to those of friendship, requests for assistance, aid to the needy, advice, and advancement of thy hobby; yea, and thou art obligated to repel any who wouldst transgresseth upon these commandments.
22. If thou doth promise to reply to a message and thou doth not, then surely thou shalt spill liquid into thy keyboard and shorteth out thy central processing unit.
23. Thou shalt not giveth any false information when applying for membership to thy BBS, for Verily it is Written that whosoever shall do so will surely be found out and thy welcome on all boards will thus be denied forever and ever.
24. Thou shalt log on properly and in accordance with thy SysOp's wishes.
25. Thou shalt observeth BBS time limits.
26. Thou shalt not upload "worm" or "trojan" or "virus-ed" programs. Amen and Amen, for to do so wilt bringeth foul curses upon thee and thine, thy children, and thy children's children yea unto the 7th generation. So shalt it be.
27. Thou shalt not ask stupid questions that art already fully explained in BBS bulletins and instructions.
28. Thou shalt not exchangeth commercial software thru thy BBS.
29. Thou shalt not violate any applicable state/federal/local laws and regulations affecting BBS telecommunications, or thee will face the wrath of thy judicial system.
30. Thou shalt not hack.



"You've never used a computer before, have you?"

BBS

Hoosier Users Group
Baud rate: 300/1200/2400
On Line 24 Hours Daily

317-782-994A

Now with a Hard Drive
40 MEG ON LINE

DISCLAIMER

This newsletter is brought to you through the efforts of the officers and members of the HOOSIER USERS GROUP. Every member is encouraged to submit articles.

If you have an article you would like to share with the other members mail it to:

Bryant Pedigo
6461 N. Sherman Drive
Indianapolis, IN 46220

Opinions expressed are those of the author and not necessarily those of the HOOSIER USERS GROUP.

15 WAYS TO WRECK AN ORGANIZATION!

1. Don't attend meetings, but if you do, arrive late.
2. Be sure to leave before the meetings adjourn.
3. Never say anything at a meeting; wait until it's over.
4. Vote to do everything at a meeting; but never do anything.
5. Find fault with officers and fellow members.
6. Take no part in organizational affairs.
7. Never invite new people to join.
8. Sit in the back and gripe with other members about the meeting.
9. Get all you can, but never give anything.
10. At every opportunity, threaten to resign and try to get others to do the same.
11. Talk cooperation, but never cooperate.
12. If asked to help, never have the time.
13. Never accept an office, but criticize anyone who does.
14. If appointed to a committee, never give time or service.
15. Don't do any more than you have to, and when others willingly, and unselfishly use their ability to help the cause along, howl because the organization is being run by a clique.

Author unknown

Feedback

OPA writes open letter

This letter is addressed to the public, and is being mailed to all of our customers which have outstanding orders from us (OPA), to selected user groups, and to MICROpendium.

Anyone who receives this letter is free to pass it on to friends and/or user groups. Newsletter editors are welcome to publish it, if they so wish.

The aim of this letter is to address a number of false rumors regarding our company (OPA), and to talk about our current state of affairs regarding outstanding orders and the reason a lot of these rumors have gotten started in the first place.

After reading this letter, remember OPA is always reachable by phone any day of the week during the hours 8 a.m. through 11 p.m. EST at either 416-960-0925 or 416-963-TITI.

We also monitor at least once a week our BBS at 416-921-2731, plus the DELPHI, USENET and Fido TI ECHO networks for any mail to us, and reply as best and often as time permits us.

If the below information is not a satisfactory explanation, we are fully willing to discuss your order either by phone or letter, and I am sure we can come to some agreement.

Some of the rumors that have been floating around recently about us, we feel deserve to be shot down. They range from saying "we NEVER made any of our products and just out to rip the public off," to "we NEVER have shipped a product by mail and just deal with us in person."

However, as most of you know, we have thousands of very satisfied customers and of course we have shipped hundreds of our products by mail since we started in business back in 1985.

We can, however, understand where some of these rumors are coming from

when you look at the following facts about which we feel the TI world needs to know. We would like to continue in the TI marketplace, but doing so is hard since the TI world is such a close-knit community that having just one unsatisfied customer reduces the total amount of orders.

The amount of daily orders and satisfying the needs of our customers is our main concern. Unlike other companies in the TI world, our customers have been the main source of our daily expenses and overhead plus employee salaries, because OPA is, most likely, the *only* company doing business in the TI world where the owners devote themselves exclusively to the enterprise and are independent of a salaried external job.

This means that a steady and increasing amount of orders is the only source of revenue to support ourselves and support future product developments. Again, because we have no outside job to cover our expenses, our product R&D cost has to be factored into the selling price of our products, and we still must stay competitive in the TI world.

We have been doing this very well since we started OPA in 1985, but over the last year we have run into a number of problems which have caused a tremendous shortfall of cash and lack of time in finishing some of our products.

Below you find a list of our products (if you wish, please contact us, for our latest catalog) each with a short explanation of their problems (if any).

• Phoenix 2001 Software: All of this line of software is being shipped and in stock. This currently includes TASS, DISKODEX and RECALLIT. One new product in final beta-testing will be added shortly.

• 9T9 Library & Assembly SIG: Since these programs come out of local 9T9 Toronto User's Group they are also available, in stock and being shipped. New Assembly SIG programs are being written and designed at the SIG meetings every Wednesday wherein OPA donates its time helping interested TIers in learning Assembly programming, hoping that new and great programs will be developed which will use more of the power of our expanded TI systems now available today.

• POP-Cart: Of the two versions of this product, the pre-programmed 256K version has been selling well and is in stock and being shipped. As for the custom version, finishing touches to customizing software that we use in making up your POP-Cart order are being scheduled; also, cash for the larger 512K EPROMs is being looked into. We plan within three weeks to have this financial problem solved; by that time, the customizing software will be finished, at which point we will start shipping the "custom" version to those with outstanding orders.

• TIM & SOB: This has been our best selling product, but also the one with the most outstanding orders, currently over 30. Being our best selling product, we indeed want to continue selling and shipping more of them. The first production lot sold so fast, we were quickly out of stock. This occurred during the time in which our own expenses and overhead were higher, and the R&D and production start-up costs of this product were also hitting us *hard*. This left us a major shortfall of cash, and incoming orders went towards things like heat, rent, food, and minimal necessities. Without these orders we would not be here today at all. We are very thankful that *we* were able to continue. The result was that we filled fewer orders on time because of part shortages, since we were unable to make the required economical bulk purchases. To alleviate this situation we had been arranging for an outside investment of money for the last six months, and as of today a solid group of investors has come with us, and will be recapitalizing the company as soon as the contract is finalized. Within three weeks, we'll be able to start rebuilding our stock of needed parts, and to begin filling the outstanding orders no later than mid-December 1992.

• ROS 8.14 & ROS_9: ROS 8.14 is still available through us, or through Bud Mills software. As for ROS_9, we have a couple of orders for it, but as of this date the software is not finished enough for EPROM use. We plan in the meantime to ship out a beta version on disk to those with outstanding orders for ROS_9 and over the next month, do tremendous testing and finishing of the ROS_9. I am sure you will be pleased with the final result. We know that ROS_9 will become the choice for most Horizon and RAMBO owners in the near future. We will release the beta version on disk shortly as ROS v8.38, containing 90 percent of new features of ROS_9.

• RAMBO/Horizon Upgrade: This product has been selling well through Bud Mills Services, but is also available directly through us and is in stock. Currently we have *no* orders for it, and have not for some time now.

• Morning Star RAMBO: This product was developed because the Morning Star memory card *never* had a DSR. Since it works the same as RAMBO (hardware-wise), we wrote a RAMBO compatible interface for it. It is in stock, but since I only know of two other people with one, besides ourselves, we doubt we'll see any orders for it.

• RAMBO Developer's Package: Until further notice, we are not going to produce this package, so we are going to return any current or future orders for it. Reasons for stopping it are too complex to go into in this letter, but anyone is welcome to call or write us about this wonderful product. In short, it has to do with protection of our work on it and the whole RAMBO project, and the direction we have been seeing our "RAMBO" work being used by other companies.

• GPL Programming Package: When we made up the catalog, since we never before published a 150 page manual we didn't envision the amount of work needed to put this package together, and with our other challenges, and time being rapidly used on meeting them, we just could not get around to finishing it. But we plan to do so as quickly as possible, and the recent addition of our own in-house photocopier will certainly reduce time spent on this and other printing needs. We will fill the few outstanding orders for this once the printing of the manual starts.

• Geneve EPROM Upgrade: We have a number of orders for this wonderful product, but have ran into a series of minor annoying problems in filling them, including our Geneve which was out of service for a few months (it seemed like years), and only recently got on-line again. We hope to schedule final work on this project within five weeks, and to start shipping soon after. If need be, we could ship a beta-version out now, but we prefer to do final bug-testing and finishing touches, since replacing EPROMs is costlier than disk update.

If you have any more questions, you're welcome to call us or write us any time at: OPA Oasis Pensive Abacutors 432 Jarvis St. #502 Toronto, Ont. Canada M4Y-2H3; 416-960-0925 or 416-963-8484 from 8 a.m. to 11p.m. EST.

Gary Bowser
Toronto, Ontario, Canada

HOME CONTROL 99 REVIEW
by William M. Lucid,
Hoosier Users Group
from a review by Ken Gladyszewski,
Northcoast 99'er

Home Control 99 by Lake Software (formerly Eagle Software) is a disk based 20K program with some assembly language routines, used to interface the TI with the X-10 POWERHOUSE.

The X-10 POWERHOUSE is an energy management system that allows one to control anything in the house which runs on electricity. It does this by sending signals over normal existing house wiring to remote modules into which lamps, etc. are plugged.

The X-10 POWERHOUSE system and Corcomp's Home Sentry Interface Cartridge were reviewed in MICROpendium (Dec. 86) and by OH-MI-TI (Jan. 87). The latter was reprinted in this newsletter (Feb. 87). Please see these articles for a more through explanation of the X- 10 POWERHOUSE system. The cartridge software as described in these articles requires only the TI console, TV, and special cable, but does not provide any additional features (files) when used with an expanded system (Disk, XBasic, Expanded memory, and RS232).

Home Control 99 uses text exclusively to an advantage, emulating the capability of X-10's software for the IBM. The user types any amount of locations and device descriptions up to the controller limitation of 256 devices. In comparsion, the cartridge allows only 14 choices of location and 9 choices of device types for a total of 126.

Using the software, the controller is programmed for up to 128 timer events. Each timer event consists of an on, off or dim command for up to 16 devices within a single housecode (while module allows for only one device per timer event).

The best feature of this program is the ability to save collections of timer events to disk as a file. This allows one to have a file for vacation, summer, winter, etc. These files can be edited, printed, and downloaded to the controller.

The X-10 POWERHOUSE controller, including required RS232 cable can be purchased from Radio Shack (Plug'n Power Computer Interface [cmc] 61-2617 $69.99)

Home Control 99 is provided on a SSSD disk with documentation on hardcopy, including instructions on how to rewire 2 wires in the IBM cable. The program sells for $10.00 and is available from:

Lake Software
c/o Paul Wheeler
1269 East 348 Street
Eastlake, Ohio 44095
(216) 946-4995

HUG OFFICERS

President	Gregory Larson	783-4575
Vice Pres	Bryant Pedigo	255-7381
Secretary	Dan H Eicher	241-994A
Treasurer	Fred Edstrom Jr	898-7300
Librarian	Bryant Pedigo	255-7381

TOURNAMENT SOLITAIRE

Reviewed by Jim Peterson

Tournament Solitaire is a collection of seven different card solitaire games on disk. You can select any of the games from the load menu, or elect to play all seven in sequence as a "tournament", hence the name.

The games were programmed by William Reiss in Extended Basic with assembly links, and the disk is available from Asgard Software (P.O. Box 10306, Rockville MD 20849) for $14.95 plus $2.50 for shipping and handling (U.S. and Canada; $7.50 for airmail elsewhere; 7% additional for credit card orders). The disk is accompanied by a very neatly published 7-page manual of instructions.

As a programmer, I can appreciate the skill and the effort that went into writing these seven programs. The graphics are all that can be done on the TI in Extended Basic, colorful and legible. The programming logic appears to be flawless - in none of the games was I able to make an illegal move, nor was any legal move refused. The manual is well written, although a bit sketchy - I still don't quite understand how to play the "Corners" game.

The seven games are Golf, Pyramid, Klondike, Canfield, Calculation, Pile Up and Corners. As far as I know, only two of these have previously been programmed for the TI - Klondike by Schererville and under the British name Patience by Gadget Man, and Pyramid by Regena. Of the others, Canfield was the only one I had ever heard of.

To evaluate computerized card solitaire games, one must ask two questions - how do they compare with Walt Howe's Chainlink Solitaire, and are they easier and more enjoyable to play on the computer than with a deck of cards?

The first question is perhaps unfair, because I consider Chainlink Solitaire to be the best "brain game" ever programmed on the TI-99/4A.

As for the second, the shuffling and laying out of the cards is far quicker than could be done manually, thanks to the assembly link. Thereafter, action slows down. Moving cards from one stack to another is accomplished by using the arrow keys to move a cursor to the card to be moved, pressing the space bar to select it, using the arrow keys to move to the position it is to be moved to, and pressing the space bar again. Cards on the stack are turned over by pressing the Enter key, and some games also use other keys. The method of playing is the same for all the games, which makes it easier to play a tournament.

Many people would probably much rather use the joystick than the arrow keys. Personally I would very much prefer to simply select a numbered pile by pressing a number key, as Chainlink Solitaire is played.

In spite of the cumbersome method of play, I did find these games to be very entertaining and addictive, and I spent a good deal of time playing them when I should have been doing something more productive.

I liked Pyramid, although it is one of the slowest in play, because it allows some opportunity for strategy. Its rules differ in one respect from Regena's version, which enabled me to actually beat the game once. I also managed to win at Pile Up, a complicated game with 20 piles of cards, which allows two reshuffles and a draw during the game, as well as peeking into stacks.

Calculation is an unusual game which might permit considerable strategy, but would require a great deal of study. Klondike is the well-known solitaire game - it could have been improved by automatically turning exposed cards face up. Canfield is a variant of Klondike. Golf is the fastest playing, and very addictive.

Is it worth buying? Absolutely!

DEBUGGING

by Jim Peterson

When you have finished writing a program, the next thing you should do is to run it. And, very probably, it will crash!

Don't be discouraged. It happens to the very best of programmers, very often.

So, the next thing to do is to debug it. And you are lucky that you are using a computer that helps you to debug better than some that cost ten times as much.

There are really three types of bugs. The first type will prevent the program from running at all - it will crash with an error message. The second type will allow the program to run, but will give the wrong results.

And the third type, which is not really a bug but might be mistaken for one, results from trying to run a perfectly good program with the wrong hardware, or with faulty hardware. As for instance, trying to run a Basic program, which uses character sets 15 and 16, in Extended Basic.

First, let's consider the first type. The smart little TI computer makes three separate checks to be sure your program is correct. First, when you key in a program line and hit the Enter key, it looks to see if there is anything it can't understand - such as a misspelled command or an unmatched quotation mark. If so, it will tell you so, most likely by SYNTAX ERROR, and refuse to accept the line.

Next, when you tell it to RUN the program, it first takes a quick look through the entire program, to find any combination of commands that it will not be able to perform. This is when it may crash with an error message telling you, for instance, that you have a NEXT without a matching FOR, or vice versa. And finally, while it is actually running and comes to something that it just can't do, it will crash and give you an error message - probably because a variable has been given a value that cannot be used, such as a CALL HCHAR(R,C,32) when R happens to equal Ø.

The TI has a wide variety of error messages to tell you when you did something wrong, what you did wrong, and where you did it wrong. But, it can be fooled! For instance, try to enter this program line (note the missing quotation mark). 1ØØ PRINT "Program must be s aved in:"merge format."

And, sometimes you may be told that you have a STRING-NUMBER MISMATCH when there is no string involved, because the computer has tried to read a garbled statement as a string.

Also, the line number given in the error message is the line where the computer found it impossible to run the program; that line may actually be correct but the variables at that point may contain bad values due to an error in some previous line.

If the error occurs in a program line which consists of several statements, and you cannot spot the error, you may have to break the line into individual single- statement lines. This is the easiest way to do that - Be sure the line numbers are sequenced far enough apart. Bring the problem line to the screen, put a ! just before the first ::, and enter it. Bring it back to the screen with FCTN 8, retype the line number 1

higher, use FCTN 1 to delete the first statement and the ! and ::, put a ! before the first ::, and continue. Then, when you have solved the bug, just delete the ! from the original line and delete all the temporary lines.

Pages 212-215 of your Extended Basic manual list almost all the error codes, and almost all the causes of each one - it will pay you to consult these pages rather than guessing what is wrong.

You may create some really bad bugs when you try to modify a program that was written by someone else - especially if you add any new variable names or CALLs to the program. Your new variable might be one that is already being used in the program for something else, perhaps in a subscripted array. I have noticed that programmers rarely use @ in a variable name, so I always tack it onto the end of any variable that I add to a program.

Also, the program that you are modifying may have ON ERROR routines, or a prescan, already built in. The ON ERROR routine was intended to take care of a different problem than the one you create, so it could lead you far astray - you had better delete that ON ERROR statement until you are through modifying.

The prescan had better be the subject of another lesson, but if the program has an odd- looking command !@P- up near the front somewhere, it has a prescan built in. And if so, if you add a new variable name or use a CALL that isn't in the program, you will get a SYNTAX ERROR even though there is no error. One way to solve this is to insert a line with !@P+ just before the problem line, and another with !@P- right after it.

When a program runs, even though it crashes or is stopped by FCTN 4 or a BREAK, the values assigned by the program to variables up to that point will remain in memory until you RUN again, or make a change to the program, or clear the memory with NEW. This can be very useful. For instance, if the program crashes with BAD VALUE IN 68Ø, and you bring line 68Ø to the screen and find it reads CALL HCHAR(R,C,CH) just type PRINT R;C;CH and you will get the values of R, C and CH at the time of the crash. You will find that R is less than 1 or more than 24, or C is less than 1 or more than 32, or CH is out of range.

In Extended Basic, you can even enter and run a multi-statement line in immediate mode (that is, without a line number), if no reference is made to a line number. So, you can dump the curent contents of an array to the screen by FOR J=1 TO 1ØØ::PRINT A(J);: : NEXT J - or you can even open a disk file or a printer to dump it to.

You can also test a program by assigning a value to a variable from the immediate mode. If you BREAK a program, enter A=1ØØ and then enter CON, the program will continue from where it stopped but A will have a value of 1ØØ.

You can temporarily stop a program at any time with FCTN 4, of course (the manual says SHIFT C, but it was written for the old 99/4), and restart it from that point with CON. Or you can insert a temporary line at any point, such as 971 BREAK if you want a break after line 97Ø. Or, you can put a line at the beginning of the program listing the line numbers before which you want breaks to occur, such as 1 BREAK 96Ø,97Ø,98Ø Note that in this case the program breaks just BEFORE those listed line numbers. You can also use BREAK followed by one or more line

numbers as a command in the immediate mode.

The problem with using BREAK and CON is that BREAK upsets your screen display format, resets redefined characters and colors to the default, and deletes sprites. So, it is sometimes better to trace the assignment of values to your variables by adding a temporary line to DISPLAY AT their values on some unused part of the screen. If you want to trace them through several statements, it will be better to GOSUB to a DISPLAY AT. And if you need to slow up the resulting display, just add a CALL KEY routine to the subroutine.

Sometimes, your program will appear to be not flowing through the sequence of lines you intended (perhaps because it dropped out of an IF statement to the next line!) and you will want to trace the line number flow. This can be done with TRACE, either as a command from the immediate mode or as a program statement, which will cause each line number to print to the screen as it is executed. If used as a command, it will trace everything from the beginning of the program, so it is usually better to insert a temporary line with TRACE at the point where you really want to start. Once you have implemented TRACE, the only way to get rid of it is with UNTRACE.

TRACE has its limitations because it can't tell you what is going on within a multi-statement line, and it will certainly mess up any screen display. Sometimes it is better to insert temporary program lines to display line numbers. I use CALL TRACE( ) with the line number between the parentheses, and a subprogram after everything else

```
30000 SUB TRACE(X)::DISPLAY AT(24,1):X :: SUBEND
```

Some programmers use ON ERROR combined with CALL ERR as a debugging tool, but I can't tell you much about that because I have never used it. ON ERROR can give more trouble than help if not used very carefully, and I cannot see that CALL ERR gives any information not available by other means.

Sometimes you can debug a line by simply retyping it. It is only very rarely that the computer is actually interpreting a line differently than it appears on the screen, but retyping may result in correcting a typo error that you just could not see. In fact, most bugs turn out to be very simple errors.

When you are debugging a string-handling routine, don't take it for granted that a string is really as it appears on the screen - it may have invisible characters at one or both ends. Try PRINT LEN(M$) to see if it contains more characters than are showing; or PRINT "*"&M$&"*" to see if any blanks appear between the asterisks and the string.

There is no standard way to debug a program. Each problem presents a challenge to figure out what is going wrong, to devise a test to find out what is really happening.

Don't debug by experimenting, by changing variable values just to see what will happen, etc. Even if you succeed, you will not have learned what was wrong so you will not have learned anything - and if your program contains lines that you didn't understand when you wrote them, you will have real problems if you ever try to modify the program. (Believe me, I speak from experience!)

TIME DATED
December 20, 1992
MATERIAL

May 1993

Dan H. Eicher
P.O.Box 605
Moorresville, IN 46158

HOOSIER USERS GROUP
P.O. Box 2222
Indianapolis, IN 46206-2222

Forwarding and Address
Correction Requested

APPLICATION FOR MEMBERSHIP

Below you will find an application for membership to the Hoosier Users Group. Active membership entitles you to the Newsletter, up and download on the HUGbbs, attendance and voting rights at regular club meetings, access to the HUGger Library of Programs, special club activities and special guest speakers for one year.

Make check or money order payable to **Hoosier Users Group.** Send completed application to:

HOOSIER USERS GROUP
P.O. Box 2222
Indianapolis, IN 46206-2222

please print — cut on line

Annual membership is $18.00
Due in May

New membership is prorated as follows:

May -------	April	$20.00
June ------	April	$18.50
July ------	April	$17.00
August ----	April	$15.50
September -	April	$14.00
October ---	April	$12.50
November --	April	$11.00
December --	April	$10.00
January	+ annual	$26.00
February	+ annual	$24.50
March	+ annual	$23.00
April	+ annual	$21.50

NAME _____ TODAY'S DATE _____

ADDRESS _____ APT # _____

CITY _____ STATE _____ ZIP _____

PHONE (____) ____-____

INTERESTS/
COMMENTS _____