Price $2.00

NEWS

North Texas PC Users Group

5.8

August. 1986

North Texas PC NEWS
(STARMAIL ADDRESS S1563)
Published monthly by members of North Texas Personal Computer Users Group for their use. Members each receive a free subscription; for others, price of the NEWS is $2 per copy. Members are requested to notify the Membership Chairman in writing of address changes. Send all editorial correspondence to:
North Texas PC NEWS
2025 Rockcreek Drive
Arlington, TX 76010.

Editor
John Pribyl (817)275-4109
Newsletter Exchange Editor
Tom Prickett (214)690-9087
Software Review Editor
Dick Gall (214)234-8888
Advertising Manager
Ross Carter (214)238-8638

Deadlines:
All material for publication in North Texas PC NEWS (articles and ads) must be received by the NEWS staff no later than Wednesday before the fourth Friday of the month.

Articles:
Please do not right-justify, indent or otherwise code the copy. If column alignment is critical, send along a hard copy, or written instructions. Article submission is preferred by modem (817/275-4109 or Startext 51563), or disk in ASCII format, unjustified. If you send a disk, please include a printed copy of the article to assure accuracy. Double spaced, typewritten copy is acceptable but must be received a week before the deadline.

Circulation:
North Texas PC NEWS circulation is 1040. Member distribution is 848; remaining copies are distributed to PC user groups around the country, and to advertisers, prospective members and others with common interests.

D E A D L I N E

* Copy deadline for September PC NEWS is August 20th (mid week after the August meeting). If possible, please finish your articles before the 16th and bring them to the meeting.

August Meeting - 3rd Saturday (16th)
September Meeting - 2nd Saturday (13th)
October Meeting - 2nd Saturday (11th)

---

* Deadline is now Wednesday!

...from the Editor's desk.

Since 1982, when North Texas PC User Group was first established, the newsletter has been distributed using First Class mail. The premium cost for this was approved because of quicker delivery, and the forwarding service that the Post Office affords First Class mail. This may change soon. With our newsletter mailing approaching 1000 copies, last month's postage bill was over $500. Because of the expense, we'll soon be forced to go to Bulk Mail. This will mean the deadline will be earlier because of inherent delays in delivery of Bulk Mail, and newsletters will no longer be forwarded, or returned. We don't know how many are presently forwarded each month, but we regularly see 8 or 10 returned because the "forwarding order has expired", or "moved and left no forwarding address", or other such notations. We don't want to lose you; be sure to notify Bob Russell when your mailing address changes.

Thanks to Microsoft, Morgan Computing and Westlake Data for donating door prizes last month. Charles Knight won a copy of QuickBasic, Orlando DeAcutis won Disk Toolkit, and Reagan Andrews and Bob Karlebach won two programs each: Safety Net and Pathfinder.

The following note is from the July 28th meeting of the Dallas Computer Council:

> "The new forms for children at Infomart should be filled out by the escorting adult EACH MONTH when entering Infomart. These forms simply identify minors in the building. Blank forms are available at any kiosk or booth near the front doors. Please cooperate and fill these out. Infomart has been sufficiently concerned about problems with unruly or unsafe behavior that they have said we (DCC) must eliminate it if we wish to continue using their facility."

John

This issue of North Texas PC NEWS was composed using PS Technical Word Processor by Scroll Systems. Repro was printed on a Toshiba P351 dot matrix printer in Prestige Elite and GTHIC151 typefaces.

August Programs — Charles Kroboth, Program Chairman

BORLAND

There will be three representatives attending our meeting from Borland International. David Intersomonie - Director of Products, Greg Joy - Sales Manager, and Steve Dix - Midwest Sales Rep will be attending several meetings and special interest groups on Saturday. BORLAND's schedule for our August meeting is as follows:

9:00 AM, Auditorium	Artificial Intelligence with PROLOG
10:15 AM, Auditorium	REFLEX REFLEX Workshop
12:15 PM, Auditorium	Utilities SIDEKICK SUPERKEY LIGHTNING TRAVELING SIDEKICK

Special Interest Groups
to be attended by Borland:

12:00 Turbo Pascal SIG

1:00 Artificial Intelligence SIG

2:00 Advanced Programmers SIG

☆ **DOOR PRIZES**

There will be Door Prizes at each meeting and SIG that Borland attends.

COMING NEXT MONTH

Sept. 9:00 AM Certified Accounting System III

Sept. 10:15 AM DataFlex from DATA ACCESS CORP.

OPEN SHOWROOMS AT INFOMART MARCH 15

NOVELL on the 3rd Level

A Novice's Walk Through Turbo Prolog

by Ray Quay

Those of you who are looking forward to TURBO PROLOG as an inexpensive, easy to use interactive environment for learning the PROLOG language are going to be ecstatic. Those of you who are expecting Turbo Prolog to provide a new environment to implement sophisticated PROLOG expert systems or knowledge representation systems are likely to be greatly disappointed. Those who want an inexpensive PROLOG environment to execute some fast and spiffy system level applications will find TURBO PROLOG frustrating but usable.

Borland's entry into the PROLOG arena has much the same characteristics as their original entry with TURBO PASCAL, just fancier. Borland has introduced a heavily extended subset of what most consider to be standard PROLOG. An extended subset may sound like a contradiction of terms, but in this case it is appropriate. The extensions include windowing (graphics and text), graphic primitives, string manipulation, system level hooks and operators, and a fairly complete file I/O system. In some aspects these are more advanced than Turbo PASCAL 3.0, and in others less advanced. The extensions are certainly similar to those found in the over $500 PROLOG systems (even some of these do not provide similar extensions). Turbo PROLOG, which at first glance appears to be a subset of the standard Clocksin and Mellish PROLOG, is actually a new dialect of PROLOG that has several PASCAL overtones. The following examines Turbo PROLOG, how it differs from other systems (some of which I have only passing knowledge), what are the good and bad features, and what are some of the bugs (yes there are bugs!).

Turbo PROLOG is clearly targeted to a mass market of C, MODULA, and PASCAL programmers, who are looking for an inexpensive introduction to artificial intelligence via the PROLOG language. Turbo PROLOG provides a complete PROLOG development system which includes an editor, compiler, an interpreter like run time interface, and simple source code tracing capabilities. Unlike Turbo PASCAL, Turbo PROLOG compiles to object code (.OBJ) which can then be linked to form executable (.EXE) files. The Turbo PROLOG system comes with a 221 page manual (plus 14 pages of Borland commercials) and two disks. The Turbo PROLOG system comes on one disk (almost the whole 360K!) and the second disk contains most of the examples in the manual, as well as a stand alone geographic data base query system.

The manual is divided into three main sections: tutorials, programmers reference, and appendices/index. The first 125 pages are a series (8) of tutorials which do a fair job of explaining Turbo PROLOG syntax and operation. Some examples do not run exactly as described in the manual, but none do so in a way that would be confusing. The last tutorial is several actual program examples. One, a solution to the famous "Towers of Hanoi", problem is quite elegant and exciting to watch. Using character graphics, it solves the problem and displays the moves required. When run at full speed it is blindingly fast. Those who have written recursive procedures in other languages to solve this problem will appreciate this solution.

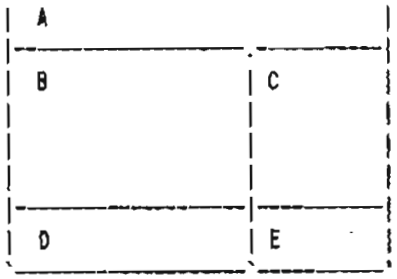
The second part of the manual is for more advanced prolog programmers. It provides a more concise and technical description of Turbo PROLOG's syntax. This section covers the compiler directives, tracing (debugging), the use of databases, flow patterns, modular programming and interfacing to other languages. This section is very concise and in some sections quite sparse. Even the most advanced PROLOG programmer will need to refer to the tutorials to fully understand Turbo PROLOG's file I/O, windowing, and graphics extensions. Borland should have spent more time here. A reference section is also provided which provides a short summary of the system's functions, the editor's functions, each system predicate, Turbo PROLOG's BNF syntax, and compiler directives. This provides a good quick reference guide while writing code.

The last section includes the appendices, glossary, and index. A rather sparse ASCII code chart (This is a Borland tradition), a numerical index to Turbo PROLOG's error codes, and a glossary are provided. One appendix is rather humorous. Appendix E "Using Turbo Prolog with Turbo Pascal" basically says in 7 lines, "not today maybe tomorrow". This appendix also gives an expected target date for the version 4.0 Turbo PASCAL ►

(rumored to create .OBJ files) as the end of the second quarter of 1987. It is also pleasing to find an index. So many systems just do not have one. This one is fair. It does not give a good multiple reference for most subjects, there are some very basic references (like stack, heap, printer) missing, and none of the predicates are included in the appendix.

The system itself is a joy to work under. The Turbo PROLOG system provides an integrated window environment for code development. The system includes a built in editor, an interpreter-like run time feature, and a source code tracer. This is very similar to QuickBasic version 2.0, and I would not be surprised if this will be the interface used for version 4.0 of Turbo PASCAL.

Each function has a dedicated window, all three of which can be on the screen at once. The main screen is divided into 5 windows. Refering to the diagram below, the largest window is the editor (B). There is also a main menu window at the top of the screen (A). A dialog window is to the right of the editor (C). This dialog area is where PROLOG code is executed in an interpreter-like fashion. The bottom of the screen is divided into two windows, the left window is a message area (D) that displays the status of the compiler, and a trace window (E) that displays the predicates being executed and the values being passed to and from each predicate during the trace mode.



Turbo PROLOG is a menu driven system, and opens up various windows to display menus, directories, and system prompts. A second editing window can also be opened to edit data bases, documentation, or any other text file.

The size and color of these windows can be adjusted with the Setup menu option. This system needs to be worked on a little. Any window can be repositioned and/or changed in size. The manual gives a short description of how to change a window's size. The cursor keys can be used to move a window or move its right or bottom border (smaller or larger). What is not clearly stated is that the upper right hand corner, once moved, remains in place. Since the dialog window is up against the right side of the screen, the whole window must be moved to the left before it can be made larger. This is not explained clearly in the manual. Each window's Color can be adjusted but the mechanism to do so is not very straightforward, and the results are not always as anticipated.

The editor is a wordstar clone similar to Turbo PASCAL, with one major difference: it cannot be customized by the user. This will drive you crazy if you've been using a customized Turbo PASCAL editor! One nice feature is that the Turbo PROLOG system has a little Artificial Intelligence built in. After you repeatedly use a sequence of keystrokes to exit, compile, and run your application, it remembers the sequence and will initiate it on recognition of the first few key strokes. This takes a bit getting use to but is a nice feature. The editor's search and replace capabilities are not quite as good as Turbo PASCAL. No U,G, or N parameters can be specified, instead a series of questions are prompted. This is rather time consuming. The search algorithms do not seem as fast, but this could be my imagination. Being able to open a second editing window on another file is nice. This allows one to not only edit a data base, but also to view other programs for ideas or structure, or even edit an include file. The block read is kind of interesting. When you specify the file to read, it opens up a window and you can specify exactly what parts of the file are to be inserted.

The language itself follows the Edinburgh syntax (ala Clocksin and Mellish) as opposed to that used with micro-PROLOG. One can use either an (IF AND .) structure or the more common (:- , .) structure. Turbo PROLOG is clearly a subset of Clocksin and Mellish. Unfortunately, Borland practically ignores that there are other implementations of PROLOG, and clearly ignores that Turbo PROLOG is not a complete implementation of PROLOG as described by Clocksin and Mellish. In fact there are only two or three places in the manual where the word PROLOG is not prefaced with the word Turbo! In all three versions of ►

Turbo PASCAL there was a section in the appendix which described the differences between Standard and Turbo PASCAL. Borland has included no such reference in this version of Turbo PROLOG. It is sorely missed. Many of the STANDARD (al cccc defacto) predicates are provided including: asserta, assertz, consult, fail, findall, isname (name), nl, not, retract, write, and ! . But then many are missing including: arg, atom, atomic, functor, mod, name, nonvar, repeat, tab, true, and var. Some of these are replaced partially by predicates unique to Turbo PROLOG like bound() and free(). Others can be written as new predicates, and some, like functor, have no equivalents. A discussion of these omissions and differences would be useful. Maybe they intend to release such a discussion in a new package called Turbo PROLOG Tutor!

The biggest difference in Turbo PROLOG is that it is a typed PROLOG. This is apparently a trend of many of the newer compiled PROLOG systems. Turbo PROLOG's types are called domains, which must be declared in the domain section. Turbo PROLOG has several built in domains that do not need to be declared: integer, real (8 byte IEEE), symbol, char, and string, and file. There is not much difference between the symbol and string domains, and they can be used in most places interchangeably. There is however a difference between the char and string domains. String constants are defined with double quotes (like basic as opposed to PASCAL) and a char constant is defined with single quotes. I am perplexed at the reason for this. Other domains, such as lists, can (must) be declared in the domains section. Each new domain can be made up of other simple domains, compound structures or have multiple definitions. A domain can also be described as a specific functor.

The typing system makes developing new general utilities difficult. Just like in Turbo PASCAL, a procedure can only pass values of the typed declared in the parameter list. With Turbo PROLOG each predicate declaration includes the domain of the parameters to be passed. This can be rather restrictive if a predicate needs to be able to handle a variety of domain types. Fortunately, Turbo PROLOG will allow predicates to have multiple declarations with different domains passed. This is not as elegant as standard PROLOG, but works. However, because all list domains must be declared, this becomes a real pain when developing general utilities that work on lists. Since Turbo PROLOG does not include predicates like reverse(), append(), and member() as part of the base predicates, the user must define them. Since integer, symbol, real, and string lists are all different domains, these predicates must be declared separately for integer, real, symbol, character, and string lists. That is five declarations for each general predicate, not counting a declaration for every user defined domain. This is not desirable. It would be better to have one reverse() predicate that reverses all lists of any domain.

Also, a predicate with multiple declarations will not work with database like predicates. For example,

```
predicate
 test(integer,integer)
 test(real,real)
clause
 test(1,2).
 test(0.1,0.2).
```

will generate a type compile error. However,

```
predicate
 test(integer,integer)
 test(real,real)
clause
 test(X,Y) :- Y = X * X.
```

will work with integers and reals.

A typical Turbo PROLOG program consists of several sections. This could include a domain section, where variable types are declared; a data base section, where predicates for a data base are declared; a goal section, where if a goal is to be achieved the rules are defined; a predicates section where the allowable arguments for each predicate are defined, and a clauses section, where each predicate is actually defined.

All predicates must be declared in the predicate or data base section. Data bases are loaded into RAM with the consult() predicate. They can then be saved back to disk. If they are too large to fit into RAM, then a file base system can be developed. Such file based data base system is not inherent in the Turbo PROLOG system, but the ►

manual includes a good example of such a system. It would have been nice to have this example on disk.

The trace option is well done. Compiler directives give a great deal of control over what is going to be traced. The results of the trace are displayed in the trace window, and the cursor is placed on the line in the edit window that is being executed. One interesting aspect of the manual is that it never explains how to get trace output to the printer. Some experimentation revealed that Ctrl-PrtSc and Ctrl-P will send this output to the printer (sound familiar?). There are two levels of tracing that can be done, a full trace and a condensed trace (shorttrace). Also, the compiler directive can indicate tracing only certain predicates. The predicate trace() and the toggle Crtl-T can be used to turn tracing on and off during execution.

Another debugging tool is the diagnostics compiler directive. This is a predicate dump, which lists each predicate, its size, variables passed, and flow pattern. This compiler directive has an intermittent bug in it. Occasionally when used, it goes berserk, printing out hundreds of odd characters. This can be a pain when you have just pressed Ctrl-Prtsc and your printer starts printing multiple formfeeds. This is a bug that should have been pick up. Even some of Borland's own sample programs cause this directive to go wild.

The dialog window provides a runtime environment that has some characteristics of an interpretive system. From the dialog window, if no specific goal has been specified in the code, any predicate can be executed. Any unbound variables passed when the predicate is evoked are printed in the dialog window with their returned value after the predicate has executed. If there are multiple responses possible, it prints the results for each response, similar to findall(). This is quite useful when developing predicates.

Turbo PROLOG has a lot of compile and runtime error messages, over 150! All are listed in appendix B, but there is very little discussion any where in the manual about what they mean and what to do to correct them. Most are very straight forward, some however are kind of obscure. Example "Can't execute a read option", "This is the first occurrence of this variable", "Loop in the flow analysis; don't use a compound flow pattern here".

The most frustrating is "This flow pattern doesn't exist for the standard predicate". All of Turbo PROLOG's standard predicates have a flow pattern which must be followed. Some make sense and others do not. The flow pattern for most of Turbo PROLOG's predicates are described in the Reference Guide section of the manual. However, nowhere are the flow patterns for Turbo PROLOG's arithmetic predicates identified in the manual, and it is some of these predicates that pose problems. Turbo PROLOG has several arithmetic predicates to manipulate integers at the bit level. For example bitand(X,Y,Z) binds to Z the value resulting from anding X and Y. However the predicate bitand(1,X,1) where X is bound to the value 3 produces the error message "flow pattern doesn't exist". This is unfortunate in that it would be useful to use this predicate to test if certain bits in a bound variable were set since and, or, xor are not provided as operators. Here is such a predicate defined in Turbo PROLOG.

```
predicates
 bitset(integer,integer,integer)
clauses
/* BITSET (X,Y,Z)  (integer,integer,integer)
  (i,i,i) True if X and Y = Z.
  (_,i,i) True if Y bits are set in Z (Y and Z = Y).
  (i,i,o) Y bits are set in X and put in Z (X and Y =
Z).
  (o,i,i) X are bits in Z not set in Y (Y xor Z = X).
*/
  bitset (X,Y,Z) :-
    bound (X),
    bound (Y),
    bitor (X,Y,A),
    Z = A.
  bitset (X,Y,Z) :-
    bound (Y),
    bound (Z),
    bitand (Y,Z,A),
    Y = A,
    bitxor(Y,Z,X).
```

Turbo Prolog is similar to Turbo PASCAL in that several system level hooks and operators are provided. These include direct access to memory at a byte and word level, direct access to ports, a generalized interrupt gate, access to string variable segment and offset pointers, access to current stack, heap, and tail size, and (unlike Turbo ►

PASCAL) the ability to evoke any DOS command string. In my version of Turbo PROLOG there is a bug in the system() predicate, which is similar to SHELL in BASICA. The manual indicates that system() should bring DOS up under the current active window. It does not, rather it commands the full screen. Also there is a major flaw in the general interrupt gate predicate bios(). Like Turbo PASCAL the AX, BX, CX, DX, SI, DI, DS, and ES registers can be passed, unfortunately the BP and Flag registers cannot. Not being able to pass the flags makes it difficult to properly use a lot of the BIOS and DOS functions which set the carry flag bit to indicate an error has occurred.

Turbo PROLOG provides a rich library of "fancy" extensions including windowing, graphics primitives, turtle graphics, sounds, and formatted printing. Some of the graphics commands have bugs. The graphics() predicate does not always respond as documented, and the pencolor() predicate can return a error message. The makewindow() predicate does not respond well in graphics or 40 column mode. Turbo PROLOG also has several amazing predicates that allow a program to evoke Turbo PROLOG's built in editor to edit any string in a dedicated window. Not impressed ! Well buried in an obscure reference is a statement that implies that a Turbo PROLOG string can be up to 64K in length! That really sparks all kinds of possibilities. I am surprised this fact was not emphasized where strings are discussed in the manual (maybe it was and I missed it?).

Most of the micro based PROLOG systems now include a library of predicates which are frequently used. These include sorts, reversing, counting, searching, set operations, and similar predicates. Turbo PROLOG examines several of these in the tutorial, but includes no general library of predicates. Thus, such libraries will either have to be acquired or be developed by hand. Once Turbo PROLOG sales numbers get reported, this will likely spark the sale of several spinoff libraries. Borland's SIG on Compuserve now has an area devoted to Turbo PROLOG. The data library on this SIG already has a couple such libraries posted.

Turbo PROLOG's compiler can create in memory executable code or .OBJ files, which can then be linked to create-stand alone .EXE files. One omission in the documentation is that Turbo PROLOG's .OBJ files must be linked with a 2.2 or higher version of LINK.EXE. The linker that comes with IBM DOS 2.0 or Compaq DOS 2.1 will not link Turbo PROLOG's object code. I had to use the IBM DOS 3.x linker to link my object files. The back of the manual clearly says that it will work with DOS 2.0 or later. I would estimate that at least 10 to 20 percent of the messages on Borland's Compuserve SIG were comments about a LINK that did not work.

Turbo PROLOG object files can be linked with other languages. This is a rather complicated subject. The manual does a good job of explaining how to write modules in other languages and call them from Turbo PROLOG, but says nothing about writing a module in Turbo PROLOG and calling it from another language. Considering the size of the executable program generated by Turbo PROLOG, calling Turbo PROLOG from another language may not be to practical, at least with in a 640K memory scheme. The following Turbo PROLOG program compiles to 51 K in 6.6 seconds, and took 1 minute and 20 seconds to link.

My novice attempt to achieve a bench mark is as follows:

```
/*  Crude Benchmark test for
    Logical Instructions Per Second
    Ray Quay
    July 1986
*/

code = 92 /* Compiler directives to maximize
heap size */
nobreak /* and speed of execution          */

domains
 intlist = integer*  /* list of integers */

predicates
 build_int_list (intlist,integer,integer)
 int_reverse (intlist,intlist)
   int_rev (intlist,intlist,intlist)
 cycle(integer,integer,intlist,intlist,integer).

goal
 write("Enter list size "),
 readint(Listsize), nl,
 build_int_list (X,0,Listsize),
```

►

```
write("Enter cycles "),
readint(Cycles), nl,

/* reset timer to 0 tenths */
time(H0,M0,S0,T0),
time(H0,M0,S0,0),

/* get start time */
time(H,M,S,T),
/* do test */
cycle(0,A,X,Y,Cycles),
/* get finish time */
time(H2,M2,S2,T2),

/* report results */
write(H,":",M,":",S,":",T), nl,
write(H2,":",M2,":",S2,":",T2), nl,
Totcycle = Cycles * Listsize,
write("LI = ",TotCycle), nl,
LIS = TotCycle / ((S2 - S) + ((T2 - T)/100)),
write("LIS = ",LIS), nl.


clauses
/* builds a list of X integers */
build_int_list ([Y | A],Z,X) :-
  Z < X -1,
  Y = Z +1,
  build_int_list (A,Y,X).
build_int_list ([X],Z,X) :-
  Z = X - 1.

/* reverse a list of integers */
int_reverse (X,Y) :- bound(X), int_rev(X,[],Y).
  int_rev ([],X,X).
  int_rev ([X | Y],Z,A) :- int_rev(Y,[X | Z],A).

/* repeats reverse E times */
cycle(A,B,C,D,E) :-
  A < E,
  F = A + 1,
  int_reverse(C,G),
  cycle(F,B,G,D,E).
cycle(E,E,F,F,E).

/* END */
```

Though I am not an expert, since the reverse predicate takes one logical instruction to reverse each element of the list, this reports a crude measure of logical instruction per second. Running this gives a result of some where between 6KLIS to 8KLIS, depending on the size of the list and number of cycles. Larger lists slow the system down considerably, I suspect because memory is scarcer and allocation takes longer the closer the system approaches its memory limits. From reports of other micro based systems that is respectable, but not extrordinary.

All in all Borland has put together another winner. Turbo PROLOG probably will not do for AI what Turbo PASCAL did for PASCAL, but it sure is going to produce some sparkling applications. I am impressed; and yes, mine will be some of those sparklers!

Ray

SWAP PC SHOP

Four lines free each month to members; 5th through 10th lines at 15 cents per word. Larger ads at commercial space rates. Send check to the Editor for words exceeding the four-line limit. Free ads are on a space-available basis. Mail ads to the Editor.

Two FASTBACK's are left over from Hard Drive Group Purchase. If you would like to purchase one, call Newton Hunt at 352-6846.

FOR SALE: 8087 chip for the IBM PC - $75. Amdek 310A amber monitor with an IBM monochrome adapter for $225. Ron Hayes - 368-1196(w), 553-0686(h).

Local Dealer - ELLIS computer software COBOL, FORTRAN, PASCAL, BASIC, PILOT, EDITOR. Please specify MSDOS or CPM. $39.95 + T & P. DCC Members: $38.50 each to your mail box. Roy Bales 214/422-1235.

MUST SELL !! One of each only. Symphony $350 QuickCode Dbase II $80 MIcrosoft Chart $100 Books 3/4 off List & many More CALL 214/370-7158.

Do you have FORTRAN programs that had statement numbers created by a random number generator? Our FORTRAN Statement Number Resequencer can fix them! Scotia Systems, Ltd. 214/987-1779.

Special Interest Program Reports

General Special Interest Group (SIG) News

A reminder that any SIG news items for this newsletter must be received by noon of the 4th Friday of the month -- regardless of the date of the next meeting.

ADVANCED PROGRAMMERs

Alison Conn of Microsoft's Language Marketing Group joined us this past month - not to give a presentation but just to observe and participate in what Neil refers to as our "normal monthly gossip session." As you might imagine, much of our conversation centered around Microsoft products and services, current and future.

Rather than a "dump on Microsoft" session, the discussion instead turned to the different paths exemplified by the programmer interfaces on Microsoft's two latest offerings, QuickBASIC Version 2.0 and C Version 4.0. QuickBASIC provides a full programming environment that supports editing, compiling and debugging all in one interactive session. The C compiler continues the more familiar batch approach but adds flexibility and capability in the form of extra 'utility' programs. The membership was split over which approach was best; the discussion amply demonstrated that the choice of an editor is a very personal thing.

One member wished that there was some mechanism by which this user group (and other user groups) could make suggestions to Microsoft, such as proposing future enhancements to their products. Alison replied that Microsoft, and particularly she, had begun to move in this direction. She was spending this summer visiting user groups across the country, and there were several proposed methods of improving communications between Microsoft and the individual software developer. She suggested that a letter, which can be shown to others in the corporation, is often a better means of suggesting these changes than are spoken suggestions. You are encouraged to write your suggestions for improving Microsoft products to: Alison Conn, Microsoft, 16011 NE 36th Way, Box 97017, Redmond, WA 98073-9717, phone: (206) 882-8664.

The discussion was not limited to Microsoft products. The price and reliability of the current crop of PC/XT/AT clones was discussed. The general opinion was that while the prices were often very low the reliability was usually even lower. Somehow, this led to more discussion of the Intel 80386 chip and the 82786 graphics chip.

We also discussed the Prolog language and Donald Knuth's new Web programming language, and what languages currently support the EGA. Almost anything is fair game for discussion at our monthly meetings; come see for yourself.

Carrington B. Dixon

APL LANGUAGE SIG

Our August meeting will be an introductory session for folks who have never used APL. John Carpenter will "begin at the beginning" and assume no prior knowledge of APL. John will explain the unusual human-oriented history of APL (it was originally intended as a human-to-human notation) which led to the current ease of use which the language enjoys.

APL handles arrays of data with ease - something most languages find extremely awkward. Flipping arrays around, rotating them, doing running totals, etc. are so easy that the user gets a feeling of control that is impossible with other languages that limit the number of dimensions an array can have, require loops, require index variables, require variable declarations, or even require a formula for each cell in the array.

Because of its simplicity, APL is catching on in corporate information centers and is becoming widespread on PC's. IBM's Information Center products, for example, are all written in APL, and APL is very heavily used by IBM internally. IBM of course sells APL for its own PC and family, but many other vendors also sell APL for the IBM PC, the Apple Mac, Atari ST, Commodore Amiga, etc.

Come on by for "An Intro to APL"!

Jim Fiegenshue

ARTIFICIAL INTELLIGENCE

At our June meeting, previously unreported, Joe Hollingsworth, a Lisp-machine hacker at TI, talked about Lisp programming from the

Special Interest Program Reports

perspective of the professional.

At the July meeting, I spoke about frame-based knowledge representation: what it is, why it is used, and how to implement it. Because of my interest in Prolog, I concentrated on Prolog implementation of a frame-based semantic network.

Several new AI SIG activities are proposed: a bibliography and a newsletter. The bibliography will concentrate on books and magazines that are readily available, rather than on compiling a list of all AI papers. Bruce Huber has volunteered to assemble the bibliography. In the AI SIG newsletter, I hope to publish information about new AI products for the PC and new books and magazines. The first newsletter will probably start in September. Anyone who has a contribution -- or anyone who would like to speak at a future meeting about AI-related activies --should contact me.

A representative of Borland International will be present at the August meeting to discuss Turbo Prolog. Since many members have experience with the product, there should be a lively discussion.

Jim Bender

ASSEMBLER SIG

Assembler is, in the final analysis, the most fundamental language for computers. Every type of compiler eventually produces machine language code. The Assembler SIG concentrates on the use of Macro Assemblers for the 8086- 8088 family of computers. Usually a given group of instructions, with examples of their use, or algorithms that make particular use of them, is covered each month.

John Wolley

BASIC SIG

There was no SIG meeting in July, since a demo of the new version of MicroSoft QuickBasic had been scheduled at the same time as our meeting.

The new version of QuickBasic has been changed with its new editor to enable the programmer to write programs which look much like Turbo-Pascal. It would provide the capability of writing structured programs without having to declare your variables. You would also be able to use commands that are already familiar to you.

If you use some of the features of the new compiler, you will not be able to run your programs in the interprerive mode. I personally prefer using the Basic interpreter for debugging with the assistance of BDS, and then compiling the program using QuickBasic for faster execution.

During the next meeting, in addition to our regular questions about Basic programming problems, I will discuss the Basic Development System by BTS. This program is such a valuable tool that I feel anyone serious about Basic should be using it.

Herb Wilson

BUSINESS APPLICATIONS

Turnout was great for our July 26th meeting, reviewing the nationally advertised $69.95 integrated financial accounting package, DAC-Easy Accounting. Participation level was high as current users of the product revealed their love/hate relationship with this penny-wise program. Also covered was DAC Software's new enhancement products aptly named DAC-Mate and DAC-Port, which add features like file viewing and printing, screen reporter, macros, spreadsheet ports and others to the accounting package. DAC-Easy Payroll was briefly discussed. Recommendations were made on which DAC products might be beneficial to the small business person and which product should not be on a software dealer's shelf.

Tune in for an exciting August 16th meeting, when a desktop organizer and a keyboard enhancer will be demonstrated. A Compaq with floppies is always available if you want to bring your most or least favorite software for show and tell. See you there.

Bruce Schubert

C Language

Allison Conn of MicroSoft presented the program for the July meeting. She demonstrated Version 4.0 of the MicroSoft compiler, which features Codeview, a C debugger of considerable power and versatility. Allison left a copy of a demonstration disk with us, and if you bring

Special Interest Program Reports

a disk to the August meeting, you may trade it for a copy.

No program is planned for the August meeting. It is time to have one of our unstructured sessions again. Join us for an interesting discussion period.

Sid Nolte

COMMUNICATIONS

The Communications SIG group welcomes all who are interested in Personal Computer communications. The SIG meeting is normally made up of a 30 minute formal presentation followed by a free form question and answer session. For those months with no formal presentation scheduled we have a free form meeting mostly related to communications.

So, bring your questions and answers and attend this months meeting.

Fred Williams

DOS

Previously known as Beginners, the DOS SIG concentrates on every phase of the DOS command structure. It is open to members of other groups, who may be in the process of changing to MS- or PC-DOS.

John Hall

GENEALOGY

In July, a general discussion covering many subjects was held. One member likes "Genealogy on Display", a public domain program. Although it is limited to 200 records, it is written in BASIC and can be expanded to include many more records. Another member says that "Family Origins" by Analytical Solutions, is a good program.

A problem: Can a modem be used to transfer information from an Apple to an IBM? It was suggested to build a database with the same parameters, then you can do anything you want with it. The question was asked shether a transfer could be made through a DIF file.

Several members expressed interest in bulletin boards. The July-August issue of AppleGram, page 10, and the Quinsept User Group Newsletter, June-July 1986 issue, both contain a list of Bulletin Boards for Genealogy. The "Soundex Coder" program listed in that Quinsept issue has a bug. If a name contains two letters with the same numerical base, one letter must be called a "null" and eliminated -- then 3 zeroes added and truncated. Example: "Benny" comes out at 1550.

Jeannie O'Neill will discuss "Recording Your Sources" at the August meeting.

Minnie Champ

GRAPHICS

This SIG is devoted to any and all means of programming and implementing graphic displays and applications on the IBM and compatible PC's. Alan Kaye's talk last month gave us an idea of the tremendous potential in this field.

Mike Durbin

N-SQUARED ANALYSIS

The meeting on July 26th was attended by the usual hard-core dozen with about 6 new faces. Greg Morris started a discussion on technical analysis by showing over 30 charts that had been created for his recent FNN appearance. Also, most of the new features of Release 4.0 (October, 1986) were discussed and explained.

It was learned during the meeting that there can be more than one way to interpret an indicator. This is especially true of momentum-type indicators. Many people use them to identify excess moves in either direction, in other words, as overbought and oversold indications. However, they can also be used as directional indications of the market being analyzed. Extreme excesses in one direction mean exceptional strength and the move is very likely to continue instead of retract, as is the normal interpretation.

At the August meeting, the discussion on technical analysis will continue. Bring ideas.

Greg Morris

Special Interest Program Reports

SCIENCE/ENGINEERING

Thanks to Dr. Levine for his presentation in the July meeting, and to Sam for coordinating the meeting for us.

In the August meeting, Dr. Thomas Madron from North Texas State Univ. will speak on "Publishing Articles in Computer Technical Journals". Dr. Madron will give some pointers on how to go about selecting articles for particular publishers. The specifics will be on some of the problems and hazards of publishing programs, particularly, statistical programs.

Dr. Madron has published articles in PC Tech Journal, Byte and a column in ComputerWorld, plus other books on computing and statistics. The April 1986 issue of PC Tech Journal contains an article by Dr. Madron, entitled: "Statistical Correlation". He has given me the program for distribution. Please bring a formatted floppy with postage and mailer or make arrangements with me to "download" from my PC at a particular time.

I have a personal request. Please do not use my work telephone. I will soon have a new home phone number and your calls are welcome at home during decent hours, or you may leave a message on my recorder. If you are on StarText, I will gladly respond via their mail facility. My mail code number is 124994.

Arlin Collins

TURBO PASCAL

Borland will be at the August meeting. We hope and expect that we will be able to have representatives at our SIG meeting.

Phil Chamberlain

Disk of the Month

Tim O'Neil

August 1986 DOM

We are entering a new generation in DOM and I hope that you will all bear with us as we reorganize this into one of the best disk libraries in the country. I have in the last week been in contact with the Houston User Group, California User Group, and Button Group. The next few months will be exciting for me and the great group of people that work on the Disk of the Month. We hope you get excited about our library too, and will continue your patronage.. The North Texas PC Group library is yours; if you would like to add software or have any ideas for improvement, call Tim O'Neil, home 267-8981, work 540-4700, both are metro numbers. -- Thanks for your help. Tim O'Neil

PC-STYLE

PC-Style (TM), "The Program That Has a Way With Words" is the August disk of the month, program and design are by Jim Button, with manual/batch files by Bill Alvernaz.

PC-Style is not a public domain program. It is copyrighted by Jim Button. The conditions under which you may copy this program and documentation are clearly outlined in the doucumentation. The registration fee is $29.95 plus $5.00 shipping and handling, from Buttonware, Inc., Bellevue, WA 98006.

What PC-Style is supposed to do (from the documentation):

*Put "punch" into what you write.
*Cut down on the use of long words.
*Add a personal touch to sentences.
*Make your writing more readable.

How does PC-Style accomplish these goals? It is through a counting process. The program identifies and counts things: sentences, words, long words, "personal" words, action verbs, and syllables. Then it computes some ratio/averages: words per sentence, long words, personal words, action verbs, and syllables per word. Then it "graphs" these ratios on standard (arbitrary?) scales, from poor to best. The upper and lower bounds for the scales can be adjusted to suit your needs or tastes.

What else can be adjusted? Well the "personal" words for one. These words are nothing more than declensions gender of the personal pronouns, "he/she," but exexclusive of "it." I hope I said that right, because I can't find my college grammer book, right now. Also adjustable is the list of "action verbs." The list of 50 verbs begin with: bear, blow and break and end with walk, wear and work. . They are all of one syllable and the longest ("strike") has only six letters. The frequency of these verbs are supposed to determine how much "punch" your writing has. The words in the two categories described above, as well as the bounds on the scales described in the previous paragraph, are maintained in one ASCII file (PCSTYLE.PRO) that can easily be modified with any editor.

This is a very easy program to initiate. There is no setup or installation. All you do is enter the program by name, PCSTYLE. This program probably should be run from drive A whenever you want to use it. Why? The documentation. It consists of well over two dozen screens/pages of information and 36 batch files. The purpose of the batch files is to display this information selectively as help screens and to automate the printing of the screens and documentation. There is nothing wrong with running this program from your hard disk, if you want to use hard disk space this way. All these files will use about 73K bytes of your hard disk.

If you believe that clear, crisp writing with short words, short sentences and "action verbs" is important, PC-Style can provide quick numerical and visual aids to check the succinctness of the documents you create or revise. Two user response forms improve PC-Style. At $29.95, it's a bargain!

H. B. Hamilton, DOM Volunteer

This month we will also offer PD FINANCE MANAGER II, Version 1.1, an exciting accounting software program, on disks 69A and 69B. ►

DISK OF THE MONTH continued

Next month: PC TICKLE.

DISCLAIMER: The North Texas PC Users Group copies these programs as a service to the club and the members of the club. We try to test all the programs, but we DO NOT WARRANT THE PROGRAMS IN ANY WAY. YOU MUST DECIDE IF A PROGRAM IS SUITABLE FOR YOUR SYSTEM AND USE. If you ask, we will tell you what we know about any program, but the final decision to buy and/or use these programs is totally yours. We will gladly and without question exchange an unreadable diskette for one of the same program.

EXCHANGE: All members of the club are encouraged to contribute copies of public domain programs to the club library. For each new diskette of software contributed, you may select any diskette in the club library in exchange. The contributions will be reviewed before credit is issued at the next meeting.

MAIL ORDERS: At this time we will not be handling mail orders. The one exception we will make to this is if we are out of stock on a diskette at a regular meeting. For this service we will have a MINIMUM CHARGE OF $2.00 FOR EACH DISKETTE. When we have more club members volunteer to help with DOM we might be able to extend this option.

PRICE: $2.00 per diskette (if the program is on two diskettes the price is $4.00)

CATALOG DISKETTES: Curently this is a two volume set priced at $4.00. This has all of the readme files from each diskette in the club library.

MEDIA: DSDD 5 1/4" Formated as 9 sector data diskettes. Public domain software only, standard full disclaimers.

AVAILABILTY: We will do our best to have all past diskettes at each meeting. DOM sales will begin at the DOM counter around 9:00, and continue until 10 minutes before the main meeting when we will close. We will then reopen after the main meeting and remain open until around 3:00.

IBM EXCHANGE NEWSLETTER: The EXCHANGE for the current month will be avaliable at the DOM table AFTER the main meeting at no charge to paid up members of the NTPCUG.

NERD ON THE STREET...

DISK OPTIONS: When are we going to see erasable optical disks? A consensus at the Fourth International Conference on the Future of Optical Memory is "let's wait & see what IBM does". Supposedly manufacturers have working prototypes but are hesitant to release them in a market that goes as Big Blue goes.

STAND UP AND BE COUNTED: Seems like only yesterday we had a splendid July meeting. Anyone notice the hundred or so Okies wandering around starry eyed? Maybe they'd like to join the Dallas Computer Council and help us real computer users vote out the nerds. For those of you not aware, we have 2 votes out of 14 on the DCC yet represent about one third of the membership. NTPCUG also has the largest attendance on user group day at Infomart. I say we run this thing like the House not the Senate.

DOS 5.0: Some people are now being shown the new DOS 5.0 from Microsoft. It is only in Alpha release at this point and will include support for the protected mode of the Intel 80286 as well as the kernal segment of the Windows operating system. Micrografx of Richardson, Texas is one of the alpha test sites. Texas Instruments is also involved with the Graphics Device Interface (GDI) portion of Windows. They are reportedly putting the GDI source code on their 34010 graphics coprocessor. This will be sold to board makers and will give them the option of putting GDI in ROM.

COMPAQ: According to Computer Retail News, Compaq will be introducing an 80386-based microcomputer in September. Operating under MS-DOS 5.1 using Intel's 32-bit processor means a shortage of software for the first few months. IBM is not expected to release an 80386 machine until Fall 1987. PC Designs, a mail order house in Tulsa, is also expected to market a 80386 AT for less than $3000.

WISH-LIST: Toshiba 3100

ATTENTION SHOPPERS!: Blue Chip Electronics, Inc. has contracted with Hyundai Electronics Industries of South Korea to produce yet another klone. The difference is the distribution channel. How about a megabuck deal with Target Stores. They will be receiving their first shipments in mid-August.

CLONE BIOS: Many network manufacturers are taking a stand against pc-clones because of incompatibilities in the networked environment. The problem generally is with the BIOS and I understand Phoenix Technologies is one of the compatible ones. Novell is in the process of approving AT clones that run with Advanced Netware 286. Approved are; Compaq Deskpro 286, Epson Equity III, Kaypro 286i, NCR Model 8, Sperry IT, Tandy 3000, and Wyse 286.

80386: Phoenix Technologies Ltd. is now shipping a PC AT-compatible ROM BIOS for Intel's new 80386 microprocessor. They are only going to a few major customers for now and those are rumored to be AT&T, Olivetti and Quadram.

McDATABASE: Borland recently announced its purchase of Singular Software, producers of a database system for the Macintosh called Interlace. Philippe will cut the price to $99.95 and rename it Reflex for the Mac.

STOCKS: Everyone seems upset that IBM earned only $1.3 billion for the second quarter of 1986. This is a 7.7% decrease over the $1.4 billion in the same period last year, but not to worry. Last year was also a drop from the previous years $1.6 billion yet they went on to a record 4th quarter for 1985. Microsoft Corporation's year end results show a 63% increase over 1985. 4th quarter earnings were up 105%.

RETAIL: According to a recent survey, 50% of the retail stores in America carry Apple Computers. 41% IBM and 24% Compaq. Entre Computer Centers reported a $5.8 million loss for it's third quarter on sales of $113.3 million. A survey conducted by Computerland's advertizing agency showed the majority of pc buyers are not happy with service given by employees. One buyer compared shopping for a computer to going to the dentist.

THOUGHT FOR THE DAY: Real men have hard disks.

Nnnnn

Room Assignments

Saturday, 16 August 1986

Check room numbers in lobby at INFOMART

Special Presentations:

9:00 - 9:55
Borland Representatives will give an Artificial Intelligence presentation... Turbo Prolog.

12:15
Borland Utilities
SIDEKICK - SUPERKEY - LIGHTNING
TRAVELING SIDEKICK

9:00 - 9:55	Room
Science/Engineering	_____
DOS	_____
Genealogy (w/Apple)	_____
Graphics	_____
BASIC Applications	_____
9:45 - 10:10	
Orientation	_____

MAIN MEETING: 10:15 - 11:45

Representatives from Borland International will present REFLEX in the auditorium.

D O O R P R I Z E S

12:00 - 12:25	Room
Orientation	_____
12:00 - 12:55	
Assembly Language	_____
APL	_____
C Language	_____
Turbo Pascal	_____
Lotus	_____

12:30 - 1:55	Room
Invest - N-Squared	_____
1:00 - 1:55	
Artificial Intelligence	_____
Business Applications	_____
Communications	_____
Databases	_____
2:00 - 2:55	
Advanced Programmers	_____
Integrated Software	_____
Framework	_____

NEW SIG: DATAFLEX _____

MEMBERSHIP APPLICATION

NEW ___
RENEWAL ___
ADDR. CHANGE ___

Name _____

Address _____

_____

City _____ State _____ ZIP _____

Phone: Home _____-_____-_____ Metro? _____

Work _____-_____-_____ Metro? _____

Send completed application with your $24 check to:
Membership Chairman, 135 Skyline Dr., Plano, TX 75074

North Texas Personal Computer Users Group, Inc.
P.O. Box 780066, Dallas, TX 75378-0066

Board of Directors

Jim Graham, Chairman
Dick Barr
Jim Janeway
Jim Hoisington
John Priboyl

The North Texas PC Users Group is a non-profit, independent group, not associated with IBM Corporation. Membership is open to owners and others interested in exchanging ideas, information, hardware, predictions, and other items related to IBM Personal and compatible computers. To join the Group, complete the application blank printed elsewhere in this issue, and send it with $24 membership dues to address shown below. A subscription to the newsletter is included with each membership.

The Group meets once each month, usually on the second Saturday. See cover for date, time and place of the next User Group meeting.

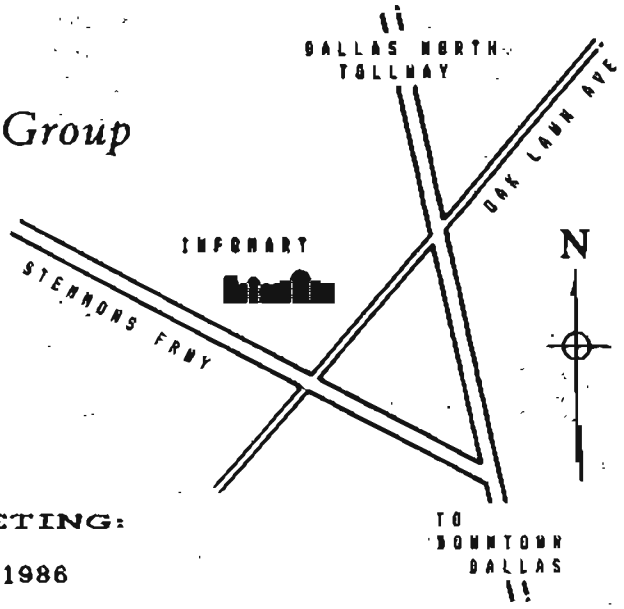
Officials:		
President	Jim Graham	(214)931-8505 h
President-Elect	Jim Hoisington	(214)245-0973 h
Program Chairman	Charles Kroboth	(214)245-4763 w
Treasurer	Joe Brophy	(214)891-8187 w
Secretary	Laura Murphey	(214)742-3233 h
Membership Director	Bob Russell	(214)422-4269 h
Disk of the Month	Tim O'Neil	(817)267-8981 h
Public Relations	Stuart Yarus	(214)867-8012 h
Special Interest Groups:		
SIG Coordinator	Phil Chamberlain	(214)243-5034 h
APL	Jim Fiegenshue	(214)539-9281 h
Artificial Intell.	Jim Bender	(214)423-3470 h
Assembler	John Wolley	(214)238-9443 h
DOS	John D. Hall	(214)495-2375 h
Business Applic.	Bruce Schubert	(214)539-7330 h
C Language	Dr. Sid Nolte	(214)233-6178 h
Communications	Fred Williams	(214)492-1315 w
Databases	Chris Morgan	(214)245-4763 w
BASIC	Herb Wilson	(214)235-3364 w
Framework	Jim Janeway	(214)349-0314 h
Genealogy	Minnie Champ	(214)341-6507 h
Graphics	Mike Durbin	(214)271-8779 h
Invest - N-Squared	Greg Morris	(214)680-1445 w
LOTUS	Susan Reyes	(214)270-3504 h
Programmers	Dr. Neil Bennett	(214)422-5673 h
Science/Engr.	Arlin Collins	(214)890-6855 w
Turbo Pascal	Phil Chamberlain	(214)243-5034 h
User Group Telephone (Answerer)		(214)242-4187
Bulletin Board SYSOP:	- o p e n -	

Payment of dues, address changes, and inquiries about membership should be directed to:
NTPCUG Membership Chairman
138 Skyline Drive
Plano, Texas 75074

Check newsletter mailing label for your membership renewal date. No separate renewal notice will be sent.

North Texas PC NEWS
2025 Rockcreek Drive, Arlington, Texas 76010

North Texas PC Users Group



NEXT MEETING:

16 August 1986