

# NORTHWEST COMPUTER NEWS

4-6 NORTHWEST COMPUTER SOCIETY  
JUN 79 P.O. BOX 4193, SEATTLE, WA 98104  
284-6109 recorded message

Single copy 47 cents / Subscription & Membership \$7.00

## Nordata Timesharing

by John P. Aurelius

It's been quite a while since anything about our Club Nordata timesharing has appeared in NCN. We have a special arrangement with Nordata to use their PDP 11/70 timesharing service at night and on weekends at very low rates. The club does the bookkeeping and hand holding as part of the deal.

Rates are 50 cents per hour and 56 cents for 4K bytes of storage per month. To qualify for this service, you must be a paid-up member of the Northwest Computer Society and you must pay a 20 dollar deposit. Roy Gillette is the Timesharing Coordinator as well as the club President, and you should contact him if you want to sign up. See him at a meeting or call 524-0596.

If you are a beginner to timesharing, here's some fundamentals. Nordata's PDP 11/70 is a very large minicomputer with an operating system designed to operate many (up to 63) jobs simultaneously, by working each for a few milliseconds at a time. You access the computer through the phone lines. If there aren't too many others making heavy demands on the machine, it seems as if you are the sole user. If the machine is busy, it works the same way but you'll get some slow responses and occasional awkward pauses. Since club timesharing is not available during business hours, the machine is not often busy when we are using it.

You have the resources of a big computer at your disposal, including essentially infinite disk storage, a very sophisticated operating system and some 'canned' software to use. You rent space on the disk in 4K byte 'clusters.' The operating system checks your storage at 3 hour intervals and charges you in little increments of time. So programs and data can be kept on the system with little effort. Our discount is not very great on storage, because of course you're using disk space during business hours as well as at night.

The machine operates all night, so night owls can get some use out of it. Nordata

does, however, routinely shut it down Saturday afternoons and evenings for backup and other maintenance. Also, if there's a problem during the day they'll patch it together until night time and — you guessed it — take it down for proper repairs at night. For all that, you can still get a lot of computing in.

This computer's fundamental language is BASIC-PLUS, which is comparable to Microsoft Extended BASIC. Of course, it isn't just the same as any of the microcomputer BASICS, any more than they are the same as each other. There is also BASIC PLUS II and Fortran. We have suffered from a lack of manuals in the past, but Roy Gillette will soon have details of how to buy manuals direct from Digital Equipment Co.; at \$5.50 for a simplified small book and \$11 for the Great Herky Reference Book.

To use the system you need a terminal with a modem and an account number from the Timesharing Coordinator. Most microcomputers can be used as a terminal by the addition of a modem and perhaps a little software. Any teletypewriter or CRT terminal can be plugged into an acoustic coupler, which needs no electrical connection to the phone line and sells for about \$200. Nordata will work with ASCII terminals at 110, 150 and 300 baud and 'mark' (no) parity. It also works with correspondence coded '2741 type' terminals, such as the Trendata 1000.

After you call the computer and log in, what do you do? It's ready to work for you in BASIC or one of the canned programs. There is a library of over 100 games (the games were gone for the last few months, but we have been assured they'll be back by the time you read this), including a fine version of Adventure. The Nordata General Manager admits to having played it for 19 hours at one stretch! There are about 6 editors, including one I wrote. There are two text processors, several sorting routines, at least two data base programs and the Club Mail file (actually a bulletin board).

continued on Page 4

## MEETINGS

**Seattle** — The Society meets at Seattle University, in the Library Auditorium, room 115. Seattle University is on 12th Avenue, between E. Madison St. and E. Cherry St. Meetings are held the first and third Thursdays of each month at 7:30 PM. The first meeting of the month usually features a formal presentation by a speaker or speakers. The second meeting is usually more informal with freewheeling discussion and problem solving.

### Thursday, July 5

John Marshall will speak on "Choosing a printer"

### Thursday, July 19

Informal meeting with no speaker scheduled. Bring your things for "show and tell."

### Thursday, August 2

John Kirk will speak on "making the TECO editor do somersaults"

**Tacoma** — The Tacoma section will meet for the summer months on the fourth Tuesday of each month at alternating locations. Contact Sam Steere (564-0875) for specific locations, or check the bulletin board at the Computerland of Tacoma store. Meetings normally have a presentation or symposium discussion.

**H-8** — The Heath group meets the first Monday of each month at the local Heathkit store, at just past 6 p.m. Contact Marty Lindal (h: 283-0806, w: 725-7111).

**CP/M & Pascal** — Meetings are held the last Wednesday of each month except December, at 7 p.m. KOMO Studio G, 4th N. & Denny Way. Contact Dick Curtiss (784-8018).

**TRS-80** — Northwest Users Teaching Society affiliate meets the second and fourth Wednesday of the month at 21814 Pacific Highway So. at the "Pine Terrace Trailer Village" recreation hall, at 7:30 p.m. Contact Dick Keller (762-4459), Walt Nash (824-4063), or Swend Miller (631-5694).

## Northstar Northwest

by Roy Gillette

There is a strong and growing body of computer enthusiasts who use and appreciate the North Star Disk Operating System and its related software. As one of these, I feel that it is only fitting that our club, the Northwest Computer Society, should include a focal point for assistance to North Star users in the Northwest. I thought first about a formal user's group, with regular meetings. But may of us are meeting'd to death already. A COLUMN in the Newsletter has significant advantages, particularly in the saving of time. Concise reading material may be scanned either quickly or slowly, as interest may dictate. Or it may be carefully studied and challenged. The careful examination of a well-designed Basic program tends to shorten my own learning curve considerably, in learning new techniques. So I believe that this column should contain as many good programming EX-AMPLES as we can get.

NORTHSTAR NORTHWEST as a regular monthly article should also present users' experience and comment on the various software PACKAGES available for that DOS. Many new and innovative packages continue to come out. But, let's face it, some of them are dogs. It would be nice to get someone else's opinion BEFORE we plunk out our hard-earned money. On the other side of that coin, when a package is really good — I'd like to know about it.

So how about it, North Star fans, are you willing to share your good ideas? your hardware and software experiences? We need you. Our capable NWCN editors will assist you in getting your thoughts onto the paper. We have some neat processes available for doing that. Don't hesitate to share.

North Star, as a set of software and related hardware, is quite unique in the small computer world. For this first article I want to call your attention to things about North Star that are refreshingly different:

1. North Star began as SOFTWARE. Its DOS and BASIC were written by Dr. Charles A. Grant and Dr. Mark Greenberg, two gentlemen who have shown remarkable insight in their long-range planning. The software was designed especially for use with the floppy disk as program and data storage. This was innovative at that time; everyone else was busy talking about cassette tape formats, how to make them compatible between users, and how to design a BASIC that would effectively use cassettes. This departure from tape and the entry into the disk world DID make it necessary to lock in on one brand of mini-floppy drive, though. The marriage of Shugart drives to the North Star Disk Operating System nevertheless proved to be a happy one. Sold as the Micro-Disk System, this free-standing drive along with its excellent software was interfaced to the SOL, the Apple, and many other S-100 computer brands. This strategy effectively captured the mini-disk market for North Star early in the game, and laid a good foundation for the continuing progress and enhancement that has been associated with North Star and its hardware from that point on.

2. North Star furnished the user with a completely INTEGRATED software package. By this I mean that the DOS was interwoven with the BASIC. We all are indebted to North Star's major competitor, MICROSOFT, for getting us started in microcomputers and providing some excel-

## LETTER TO THE EDITOR

Last month that letter to "Dear John"  
Did hit "My System" quite head-on —  
We want to see the good "how-to's"  
And not a lot of boring news  
Of wheres and hows or whos and whys  
Which one goes thru before one buys  
Computer systems off the shelf  
That one won't have to patch one's self  
'Cause now is different from before —  
We want to have a little more  
Of what's been tested, what's been done,  
If it's for work or just for fun —  
The goal that we should be pursuing  
Is learning what we could be doing.

Anita Westrum  
6/11/79

lent products. But they were unable to provide a DOS for their own BASIC, probably due to customer restrictions in their marketing plan. So by default, the Microsoft DOS was written by Structured Systems, and called CP/M. North Star, on the other hand, was able to retain full control by integrating their DOS with their own powerful BASIC, one that need not take a back seat to anyone else's "standard." Remember, a DOS is the key item to link any BASIC to the speed and versatility of multiple data files on disk. North Star was thinking ahead.

3. Another evidence of long-range planning at North Star was their introduction of a Floating Point Board. BASIC being an interpreter makes it slower than compiled languages, especially in number crunching. The FP board for S-100 computers helped considerably. In the KILOBAUD time trials (a set of benchmark tests and results published in June and October of 1977), North Star's Floating Point Board did very well.

4. We software people worry about upward compatibility, which is the ability to keep one's older programs running in spite of what the computer company does to you in operating software "improvements." North Star seems to worry about it, too. Each new release of BASIC or the DOS has supported programs written under earlier versions. Another evidence of commitment to this principle could be seen in their introduction of a double-density controller for the Horizon. Though double-sided disks would not be released until many months later, the controller for double-density was designed so it would still handle the upgraded disks at that time. Between single- and double-density, too, North Star has done their homework extremely well — our older files on disk can still be accessed and used. They can optionally be converted to double-density format, or left as is and used.

5. Documentation-wise, North Star boasts a one-inch-thick manual that explains in detail each statement, command, and error message used in their system. This book contains liberal charts, appendices, and programming examples to help make it plain. It is a professional manual, emphasis on the word professional.

continued on page 2

Northwest Computer Society  
P.O. Box 4193  
Seattle, Wa 98104

Bulk Rate  
US Postage  
PAID  
Renton, Wa. 98055  
Permit No. 282



reprinted from the New York Amateur Computer Club newsletter, May 1979. Typesetting courtesy Amateur Computer Group of New Jersey newsletter, June 1979.

## TRS-80 Tokens And Routines

by Mike Fischer

Below is a list made in exploration of the depths of the Level II BASIC firmware. Within the list, the two-digit hex number following each program statement is its "token" -- the form in which BASIC actually stores it in memory. These tokens are reconstituted into ASCII for listing and editing purposes. The four-digit hex number is the address of the routine performed when the statement is used; some, like CLS, can easily be called from one's own machine-language programs. Disk BASIC routines have addresses from 4000H. If you would like to generate the list on your system, the program to do it appears below the list.

The Program....

```
10 DIM H$(16)
20 FOR I=0 TO 15:READ H$(I):NEXT
30 DATA "0","1","2","3","4","5","6","7","8",
  "9", "A", "B", "C", "D", "E", "F"
90 I=5712
100 FOR II=0 TO 30000
200 WRD$=CHR$(PEEK(I)-128)
300 FOR J=I+1 TO 30000
400 C=PEEK(J)
500 IF C=128 THEN 1100
510 IF C>128 THEN 1100
600 WRD$=WRD$+CHR$(C)
700 NEXT J
800 NEXT II
1100 ADR=6179+2*II
1200 IF II>=60 THEN ADR=6186+2*II
1210 IF II>=87 THEN ADR=5641+2*II
1300 CODE=128+II
1400 Z2=CODE:GOSUB 5000:CODE$=Z2$
1500 Z2=PEEK(ADR):GOSUB 5000:ADR$=Z2$
1510 Z2=PEEK(ADR-1):GOSUB 5000:ADR$=ADR$+Z2$
1590 K=K+1
1592 IF K>=62 THEN IF K=87 THEN READ ADR$
1594 DATA *,4155,*,24EB,27FE,24DD,24CF,2A2F,419D
1596 DATA 0132,4176,27C9,019D,1D78,25C4,*,249F,
  2532
1598 DATA *,*,*,*,*,*,*
1600 LPRINT K;TAB(5) WRD$;TAB(15) CODE$;" - ";
  ADR$
1700 IF C>128 THEN I=J:J=31000:GOTO 700
1800 END
5000 Z2$=H$(Z2/16)+H$(Z2 AND 15):RETURN
6000 Z2=Z4/256:GOSUB 5000:Z4$=Z2$
6010 Z2=Z4 AND 255:GOSUB 5000:Z4$=Z4$+Z2$:RETURN
```

1	END	80	- 1DAE	63	FN	BE	- 4155
2	FOR	81	- 1CA1	64	USING	BF	- *
3	RESET	82	- 0138	65	VARPTR	C0	- 24EB
4	SET	83	- 0135	66	USR	C1	- 27FE
5	CLS	84	- 01C9	67	ERL	C2	- 24DD
6	CMD	85	- 4173	68	ERR	C3	- 24CF
7	RANDOM	86	- 01D3	69	STRING\$	C4	- 2A2F
8	NEXT	87	- 22B6	70	INSTR	C5	- 419D
9	DATA	88	- 1F05	71	POINT	C6	- 0132
10	INPUT	89	- 219A	72	TIME\$	C7	- 4176
11	DIM	8A	- 2608	73	MEM	C8	- 27C9
12	READ	8B	- 21EF	74	INKEY\$	C9	- 019D
13	LET	8C	- 1F21	75	THEN	CA	- 1D78
14	GOTO	8D	- 1EC2	76	NOT	CB	- 25C4
15	RUN	8E	- 1EA3	77	STEP	CC	- *
16	IF	8F	- 2039	78	+	CD	- 249F
17	RESTORE	90	- 1D91	79	-	CE	- 2532
18	GOSUB	91	- 1EB1	80	*	CF	- *
19	RETURN	92	- 1EDE	81	/	D0	- *
20	REM	93	- 1F07	82	[	D1	- *
21	STOP	94	- 1DA9	83	AND	D2	- *
22	ELSE	95	- 1F07	84	OR	D3	- *
23	TRON	96	- 1DF7	85	>	D4	- *
24	TROFF	97	- 1DF8	86	=	D5	- *
25	DEFSTR	98	- 1E00	87	<	D6	- *
26	DEFINT	99	- 1E03	88	SGN	D7	- 098A
27	DEFSNG	9A	- 1E06	89	INT	D8	- 0B37
28	DEFDBL	9B	- 1E09	90	ABS	D9	- 0977
29	LINE	9C	- 41A3	91	FRE	DA	- 27D4
30	EDIT	9D	- 2E60	92	INP	DB	- 2AEF
31	ERROR	9E	- 1FF4	93	POS	DC	- 27F5
32	RESUME	9F	- 1FAF	94	SQR	DD	- 13E7
33	OUT	A0	- 2AFB	95	RND	DE	- 14C9
34	ON	A1	- 1F6C	96	LOG	DF	- 0B09
35	OPEN	A2	- 4179	97	EXP	E0	- 1439
36	FIELD	A3	- 417C	98	COS	E1	- 1541
37	GET	A4	- 417F	99	SIN	E2	- 1547
38	PUT	A5	- 4182	100	TAN	E3	- 15A8
39	CLOSE	A6	- 4185	101	ATN	E4	- 15BD
40	LOAD	A7	- 4188	102	PEEK	E5	- 2CAA
41	MERGE	A8	- 418B	103	CVI	E6	- 4152
42	NAME	A9	- 418E	104	CVS	E7	- 4158
43	KILL	AA	- 4191	105	CVD	E8	- 415E
44	LSET	AB	- 4197	106	EOF	E9	- 4161
45	RSET	AC	- 419A	107	LOC	EA	- 4164
46	SAVE	AD	- 41A0	108	LOF	EB	- 4167
47	SYSTEM	AE	- 02B2	109	MKIS	EC	- 416A
48	LPRINT	AF	- 2067	110	MKS\$	ED	- 416D
49	DEF	B0	- 415B	111	MKD\$	EE	- 4170
50	POKE	B1	- 2CB1	112	CINT	EF	- 0A7F
51	PRINT	B2	- 206F	113	CSNG	F0	- 0AB1
52	CONT	B3	- 1DE4	114	CDBL	F1	- 0ADB
53	LIST	B4	- 2B2E	115	FIX	F2	- 0B26
54	LLIST	B5	- 2B29	116	LEN	F3	- 2A03
55	DELETE	B6	- 2BC6	117	STR\$	F4	- 2B36
56	AUTO	B7	- 2008	118	VAL	F5	- 2AC5
57	CLEAR	B8	- 1E7A	119	ASC	F6	- 2A0F
58	CLOAD	B9	- 2C1F	120	CHR\$	F7	- 2A1F
59	CSAVE	BA	- 2BF5	121	LEFT\$	F8	- 2A61
60	NEW	BB	- 1B49	122	RIGHT\$	F9	- 2A91
61	TAB	BC	- 0ADB	123	MID\$	FA	- 2A9A
62	TO	BD	- *	124	!	FB	-

## Pattern Recognition Algorithm

by Dick Curtiss

The following program demonstrates a pattern recognition algorithm. The pattern or template contains some combination of normal string segments with special parameter escape characters interspersed. The pattern recognition algorithm tries to match a string against the template by allowing any string segment (including an empty string) to match a parameter escape character. In the following example "\$" is the special parameter escape character. A normal "\$" may be forced into the template string by putting two of them together, "\$\$".

template = "\$ = \$ + \$"

string for match = "abc = d + ef"

result of match is:

parameter 1 = "abc"

parameter 2 = "d"

parameter 3 = "ef"

The template matching algorithm appears in the subroutine at line 9000 in the sample program. The special parameter escape character may be changed in line 9050. The sample program starts with a request for a template string followed by a request for a matching string. Then a match is attempted and the results are printed. A "#" response to the request for a matching string input returns to the enter template level. A "#" response to the request for a template input stops the program.

The program is written in APPLESOFT BASIC but should run on other BASICS.

Template matching subroutine notes:

Internal variables:

I, I1, I2, I8, I9

J, J2, J9

T1\$, S1\$, E\$, N\$

Input variables:

T\$ — template string

S\$ — string for matching

Output variables:

M = 0 for no match condition

M = 255 for match ok

K = number of parameters located

P\$(K) = array of parameter strings found

## RUNOFF, Simplified

by John Aurelius

Runoff is a text formatter on Nordata, which takes a rough typed file and makes pretty columns. It also can handle all upper case input and print upper/lower case text. There's a 24 page manual from Digital Equipment Corp. You can use the program, however, with much simpler instructions. The fundamental rule is simple. A line that starts with a period is a COMMAND. Any other line is TEXT.

I'm going to type this paragraph in upper case. To use this feature, tell RUNOFF you are doing so by entering a line that says ".lc" — this makes upper case text convert to lower unless you use the caret symbol (C) as a shift. My terminal is a selectric type and the cents sign becomes the caret to Nordata. Two carets together (CC) act as a shift lock, and two backslashes together act as the unlock. My terminal echoes them as a question mark.

If your terminal has lower case, remember that you must tell the computer to use it, or it will convert what you type to upper case (not a problem with Selectric terminals). After READY, type SET LC INPUT and a carriage return. Then type SET LC OUTPUT.

The only special thing we really would like for newsletter text is the ".p" command before each paragraph. Runoff stops filling, skips a line and indents when it sees this as the only text in a line. The titling stuff at the top of this text can be added by us, or even forgotten. Runoff prints a header at the top of each page, which can have two lines. The first has the title and page number, and the second has the subtitle. ".first title" just makes it print a title on the first page, which it would

continued

## Northstar Northwest

continued from page 1

sizing the finer details of how the computer will respond. So it may not be entertaining reading. The indexing plan is super. When you need to know or review something, you can actually FIND it. In all of our micro computer world I have not seen any documentation of this quality. I must say that, more than anything else, this documentation has kept me sold on North Star. They are professionals.

6. Application-wise, North Star users are way ahead. This is because so many software houses have written for the North Star system since its introduction to the market in January, 1977. The names of over 200 vendors appear in the North Star "Big Book." Each of these firms has contributed applications software of from one to ten or more programs or systems, largely in the business and accounting areas. Recently there has been a trend toward converting this software wealth so it may be used with CP/M and the TRS-80. Does this mean that new applications are slowing for the Horizon/North Star? Not noticeably, if at all. New North Star programs continue to be advertised, often with higher quality standards than we saw in the earlier days. We are talking about systems developed for the North Star, primarily — not CP/M. Keep in mind that CP/M WILL run on this system, through the conversion package by Lifeboat Associates. So North Star is the only popular multi-computer-brand DOS that allows you both CP/M applications AND its own quality applications. One has the best of both worlds.

Well, this has turned out to be quite a testimonial, as I read it back. OK, so there are some neat things about North Star, and Roy Gillette, as one software developer, recommends the Horizon and its software as the best small computer buy around, for the money invested. It can't all be good.

How about problem areas — there must be some. Well, my own chief complaint about them is the same as I have with other computer firms — I find them to be HUMAN. Particularly when it comes to production delays and software releases. At this writing (so I hear), a large number of new Horizons are being held up, due to the inavailability of one chip needed to finish the boards. So many events in this business are hard to predict that we computer people more often than not end up being late. None of us like it when our suppliers promise us a date that cannot be met. I know one thing, though — I'd rather have a quality product that is behind schedule than a poorly designed product which is easier to get due to its superior marketing strategy. How about you?

The other two complaints one hears about North Star really apply rather to minifloppies in general, as they compare to the 8-inch disks. Owners of systems with the larger drives point out that minifloppies lack the speed and storage capacity of their big brothers. That's true. So why have the smaller disks proved to be so extremely popular? Perhaps it is because, while the 5 1/4 diskettes are a quantum leap ahead of using cassettes (for speed and versatility), they still retain "tiny" appeal. People like smaller things these days (smaller calculators, smaller cars, smaller desk-top computers). And if many small business systems have been successfully implemented around only two single-density North Star drives (they have), the picture is definitely improving, from that point on. The trend is to design and market drives that will pack more and more data on these little disks. When we recently went from single-density to double, the on-line storage upped from 180k to 360k from two drives. Access speed also became twice as fast. The double-sided, double-density disk drives (to be released soon, I understand) will double it again. Byte magazine in its May, '79 issue predicted that 5 1/4-inch drives in 1980 will

be able to access one to two million bytes each! So I guess we are not hurting, in our ability to upgrade capacity.

Could North Star adapt its Disk Operating System to work with the 8-inch size drives? Undoubtedly they could, and a Texas firm is already selling just such a conversion. But those people at North Star are foxy. Make no mistake about it — they have a long-range sales plan. Who knows, perhaps they'll come out with an 8-inch HARD disk, like Cromenco, to meet the upper end users who will pay some \$8,000 for a disk with 11 million unformatted bytes of storage. But most of us will continue to use the little floppy critters. With the North Star operating software, they can do a lot.

Here is a little program that would be useful if you had just purchased a new CRT terminal and wanted to test out what all of the keys would transmit to the computer. It introduces the INCHAR\$ statement that came out with release 4, a powerful tool that I recommend you use in data entry programs.

```
100 REM chartest 6/5/79 R. Gillette
110 REM
120 !"Enter any characters you wish from
  the keyboard except whatever"
130 !"keyboard LOCK sequence your terminal
  provides (You'd have to flip"
140 !"the power switch off and on again to
  get out of THAT one. This"
150 !"routine displays the decimal value of
  each character including"
160 !"control characters, and prints it if it is
  printable. To EXIT"
170 !"press RETURN"
180 REM
200 T$ = INCHAR$(0)
210 !ASC(T$), " = ", CHR$(T$)
220 IF T$ = CHR$(13) THEN 230 ELSE
  200
230 !"DONE . . ."/!
240 END
```



```

50 DIM P$(10): REM PARAMETER RETURN          9230 GOTO 9200
JRUN
ENTER TEMPLATE                                100 PRINT "ENTER TEMPLATE";
?#=#+#                                         101 PRINT
?#=#+#                                         110 INPUT T$
ENTER STRING FOR TEMPLATE MATCH               120 IF T$ = "*" THEN STOP
?ABC=DEFG+ZYXSTW 123                          130 PRINT
PARAMETER (1) = ABC                            200 PRINT "ENTER STRING FOR TEMPLATE MATCH";
PARAMETER (2) = DEFG                           201 PRINT
PARAMETER (3) = ZYXSTW 123                     210 INPUT S$
ENTER STRING FOR TEMPLATE MATCH               220 PRINT
?I = I + 1                                     230 IF S$ = "*" THEN 100
PARAMETER (1) = I                               300 GOSUB 9000
PARAMETER (2) = I                               310 IF M = 0 THEN GOTO 500
PARAMETER (3) = 1                               315 IF K < 1 THEN 350
ENTER STRING FOR TEMPLATE MATCH               320 FOR N = 1 TO K
?PDQ = 493                                       330 PRINT "PARAMETER (";N;") = ";P$(N)
NO MATCH *****                              340 NEXT N
ENTER STRING FOR TEMPLATE MATCH               350 PRINT
?#                                               360 GOTO 200
ENTER TEMPLATE                                500 PRINT "NO MATCH *****"
?#                                               510 PRINT
?#                                               520 GOTO 200
ENTER STRING FOR TEMPLATE MATCH               9000 REM TEMPLATE MATCHING ALGORITHM
?#                                               9050 E$ = "$": REM PARAMETER ESCAPE CHARACTER
ENTER TEMPLATE                                9070 N$ = CHR$(0): REM END OF LINE CHARACTER
?#                                               9080 I = 0:J = 0:K = 0
?#                                               9090 I9 = LEN(S$):J9 = LEN(T$)
ENTER STRING FOR TEMPLATE MATCH               9100 GOSUB 9700: REM I=I+1 S1=S(I)
?#                                               9110 GOSUB 9800: REM J=J+1 T1=T(J)
?#                                               9120 IF T1$ < > E$ THEN 9400
ENTER STRING FOR TEMPLATE MATCH               9130 GOSUB 9800: REM J=J+1 T1=T(J)
?#                                               9140 IF T1$ = E$ THEN 9400
?#                                               9150 I1 = I: REM PARAMETER STRING ANCHOR
ENTER TEMPLATE                                9200 IF S1$ = T1$ THEN 9300
?#                                               9210 IF S1$ = N$ THEN 9600
BREAK IN 120                                   9220 GOSUB 9700: REM I = I + 1 S1=S(I)
JLIST
9300 I8 = I
9305 IF T1$ = N$ THEN 9480
9310 I2 = I:J2 = J
9320 GOSUB 9700: GOSUB 9800
9330 IF T1$ < > E$ THEN 9340
9331 GOSUB 9800
9332 IF T1$ = E$ THEN 9340
9333 K = K + 1
9334 P$(K) = MID$(S$,I1,I8 - I1)
9335 GOTO 9150
9340 IF S1$ = T1$ THEN 9350
9342 I = I2:J = J2 - 1
9344 GOSUB 9700: GOSUB 9800
9346 GOTO 9200
9350 IF T1$ < > N$ THEN 9320
9360 GOTO 9480
9400 IF S1$ < > T1$ THEN 9600
9410 IF T1$ < > N$ THEN 9100
9420 GOTO 9500
9480 K = K + 1
9490 P$(K) = MID$(S$,I1,I8 - I1)
9500 M = 255: RETURN : REM MATCH OK
9600 M = 0: RETURN : REM NO MATCH
9700 I = I + 1
9710 IF I > I9 THEN 9730
9720 S1$ = MID$(S$,I,1): RETURN
9730 S1$ = N$: RETURN : REM END OF LINE
9800 J = J + 1
9810 IF J > J9 THEN 9830
9820 T1$ = MID$(T$,J,1): RETURN
9830 T1$ = N$: RETURN : REM END OF LINE
    
```

Continued from Page 2

otherwise omit. If you have tabular material in your text that should not be filled, put in a ".no fill" command. Put a ".fill" command in to handle regular text after the table. To skip 2 lines without indenting, use a ".skip 2" command. To indent text 5 spaces, use ".indent 5", and ".indent 0" to restore the margin.

There's much more to RUNOFF, but you can do a lot with only this much.

Membership in the Northwest Computer Society and attendance at meetings are open to anyone interested in personal computers or computing. Membership dues are \$7.00 per year. Membership begins immediately upon payment of dues. Memberships paid in the first three months of the year, or in the last three months of the previous year, come due again as of the first of the next year without dues adjustment. Memberships paid in the middle six months of a year are charged \$10 and come due the second January away.

1979 NWCS Officers:

- Roy Gillette President
- John Kirk NCN Editor
- Buzz Towne Treasurer
- John Aurelius Secretary
- John Marshall Program Chairman
- Ken Berkun Board Member-at-large
- Sam Steere Board Member-at-large

```

.ps 58,38
.title NORTHWEST COMPUTER NEWS
.subtitle RUNOFF, SIMPLIFIED
.first title
by John Aurelius
.p
Runoff is a text formatter on Nordata, which takes a rough typed file and makes pretty columns. It also can handle all upper case input and print upper/lower case text. There's a 24 page manual from Digital Equipment Corp. You can use the program, however, with much simpler instructions. The fundamental rule is simple. A line that starts with a period is a COMMAND. Any other line is TEXT.
.p
.lc
¢I'M GOING TO TYPE THIS PARAGRAPH IN UPPER CASE. ¢TO USE THIS FEATURE, TELL ¢¢RUNOFF?? YOU ARE DOING SO BY ENTERING A LINE THAT SAYS ".LC" - THIS MAKES UPPER CASE TEXT CONVERT TO LOWER UNLESS YOU USE THE CARET SYMBOL (¢) AS A SHIFT. ¢MY TERMINAL IS A SELECTRIC TYPE AND THE CENTS SIGN BECOMES THE CARET TO ¢NORDATA. ¢TWO CARETS TOGETHER ( ¢ ¢) ACT AS A SHIFT LOCK, AND TWO BACKSLASHES TOGETHER ACT AS THE UNLOCK. MY TERMINAL ECHOES THEM AS A QUESTION MARK.
.uc
.p
If your terminal has lower case, remember that you must tell the computer to use it, or it will convert what you type to upper case (not a problem with Selectric terminals). After READY, type SET LC INPUT and a carriage return. Then type SET LC OUTPUT.
.p
The only special thing we really would like for newsletter text is the ".p" command before each paragraph. Runoff stops filling, skips a line and indents when it sees this as the only text in a line. The titling stuff at the top of this text can be added by us, or even forgotten. Runoff prints a header at the top of each page, which can have two lines. The first has the title and page number, and the second has the subtitle. ".first title" just makes it print a title on the first page, which it would otherwise omit. If you have tabular material in your text that should not be filled, put in a ".no fill" command. Put a ".fill" command in to handle regular text after the table. To skip 2 lines without indenting, use a ".skip 2" command. To indent text 5 spaces, use ".indent 5", and ".indent 0" to restore the margin.
.p
there's much more to RUNOFF, but you can do a lot with only this much.
    
```

The Northwest Computer News is produced monthly by the Northwest Computer Society, P.O. Box 4193, Seattle, WA 98104. Subscription is a benefit of membership in the Society and not currently available otherwise to individuals. Other clubs who issue newsletters are encouraged to exchange subscriptions.

1979 NCN Staff:

- John Kirk Editor
- John Aurelius Editor Emeritus & Unclassified Section & Paste-up
- Gale Sherry Photography & Paste-up
- Ken Berkun Paste-up & Contributing Editor
- Joe Felsenstein Contributing Editor

Upcoming deadlines: July 7, August 4, Sept. 1 for written copy. The week following each of these dates for photo-ready commercial ads.

Any content that is specifically marked "Copyright ©" may not be reproduced without written permission of the author. All other content may be reproduced for non-commercial purposes, provided that the author and Northwest Computer News are given credit.





## Nordata Timesharing

Continued from Page 1

To access the mail file, for example, type:

```
RUN (7,16)MAIL
```

The program prints the latest three entries in the message index, then asks if you want to continue. You can read any or all messages from any account. You can add or delete messages from any Club account.

Most of this newsletter is written on Nordata. I use my own editor program, (240,53)EDIT3. I wrote it because I wanted one simple enough to be understood with a one page manual. This lets me compose from the keyboard, correcting my typing boobos as I go and ending up with a clean rough typed text.

Then \$RUNOFF, a canned Nordata program, is called to print the text in the proper format, 38 characters wide for paste up in the newsletter or 60 characters wide double spaced for typesetting. There are commands which must be inserted in the text to make RUNOFF work. They all begin with a period, and the program treats any line beginning with a period as a command and not text. I have a copy of the RUNOFF manual, and will be happy to copy it for anyone who promises to write at least one article for NCN (hint, hint). Drop me a line at the club P O Box. RUNOFF also allows you to enter all upper case text, using the up arrow (caret) character as a shift or two up arrows as a shift lock. Two backslashes cancel the shift. So even if you don't have lower case, you can enter items for the newsletter and we can get normal text.

Data base programs make it possible to generate tabular data files using your own headings and formats. You can add, update, delete or sort data with ease. They're excellent for files like mailing lists, as one example. The club mailing list is kept on a rather simple data base program I wrote, (240,53)ANSWER. As new members join I add their names and addresses to the bottom of the file. When there are many new names, I tell the program to SORT BY NAME. After the file is sorted, I get a hard copy print out with the LIST+ command. The plus sign means to print the record number in front of each record. This number is just the record's position in the file.

To correct an entry, for example if someone has moved, I look for the name on the list and note the record number, for example, 247. Then I tell the program 247 CHANGE STREET. It responds by printing the old address and asking for input. I type the new information and it's put into the right place in the file. To print labels when the newsletter is to be mailed, I have it SORT BY ZIP. There's a WIDTH command in the program which makes it possible to get printout as wide as desired - it won't put a field on the line unless there's room for the whole field. Since the NAME and STREET fields are 21 characters, a 22 column width forces them to print on separate lines. Then I cheat (slightly): I add two PRINT statements to the code so it will put two blank lines after each record.

There is a much fancier data base program (actually a system of programs) on the Nordata computer. This is called RAP and it has a 64 page manual. The manual is available from the club for the cost of copying. For more on data bases, see Joe

Felsenstein's columns in NCN, issues 4-1 and 4-2.

You can do very sophisticated things with the Nordata system, or you can just use it to play Star Trek. It is an alternative to owning a microcomputer, and it also is a nice supplement to one's micro. If you are already using Nordata, I invite you to try my editor and data base programs. They can be run from any account, and they are self documenting.



## Get the Most Out of Your Computer with Creative Computing

**Software and Applications.** Pragmatic, well-documented programs with complete listings on data base systems, word processing, communications, simulations, investment analysis, games, music synthesis, computer art, business functions, building control and more.

**System Evaluations.** In-depth, probing evaluations of personal and small business systems every issue. Non-sense reviews of software from independents as well as the majors.

**Regular Features.** Operating Systems Q and A. Columns on the TRS-80, Apple and PET. Book reviews. Programming techniques. Short programs. Computer games. New products. Even a dose of fiction and foolishness.

**We guarantee** that Creative Computing will help you get more out of your personal, school, or business computer or we'll give you your money back!

3 years \$40 (Save \$32 over retail price)

2 years \$28 (Save \$20 over retail price)

1 year \$15 (Save \$9 over retail price)

Foreign:  Surface add \$9/yr.  Air add \$24/yr.

To order, send payment or bankcard (Visa or Master Charge) number and expiration date with your name and address to Creative Computing, Attn: Bob P.O. Box 789-M Morristown, N.J. 07960.

Save Time! Phone bankcard orders toll-free to:

**800-631-8112**

(In NJ call 201-540-0445)

**creative computing**

P.O. Box 789-M, Morristown, NJ 07960

### \*\*INSTRUCTIONS FOR EDIT3\*\* 27 MAY 79

MANY OF THE COMMANDS REFER TO A RANGE OF LINES, BEGINNING AT THE CURRENT OR NEXT POINTER POSITION AND EXTENDING DOWN THE NUMBER YOU SPECIFY. FOR EXAMPLE, 'SEA 6' WILL SEARCH SIX LINES, BEGINNING WITH THE NEXT LINE. THE ENTIRE TEXT IS SPECIFIED BY 'ALL'; I.E., 'PRI ALL'.

LINE ZERO IS ALWAYS EMPTY - IT IS USED TO ACCESS THE TOP OF THE TEXT FOR COMMANDS LIKE 'INSERT' AND 'PRINT'.

THE CURRENT LINE NO. WILL BE PRINTED, FOLLOWED BY 'E?'  
THE EDITING COMMANDS ARE:

TOP - MOVE POINTER TO TOP OF TEXT (LINE 0)  
GOTO - MOVE POINTER TO SPECIFIED LINE NO; EXAMPLE 'GO 53'  
MOVE(OR - MOVE PTR. UP/DOWN A NUMBER OF LINES. EXAMPLE 'MOV 2'  
NEXT) MOVES DOWN TWO LINES; 'MOV-2' UP TWO (DEFAULT=1).  
BOTTOM - MOVE POINTER TO BOTTOM OF TEXT

INSERT - INSERT FROM KEYBOARD - NEW TEXT IS INSERTED BELOW THE CURRENT LINE. TERMINATE WITH A LINE HAVING NO TEXT

LOAD - INSERT FROM FILE BELOW THE CURRENT LINE.  
EXAMPLE: 'LOA TEXT.RNO'

PRINT - PRINT A NO. OF LINES (DEFAULT=1), BEGINNING WITH NEXT

DELETE - DELETE A RANGE OF LINES (DEFAULT=1)  
'DEL ALL' WILL CLEAR THE WORKSPACE FOR NEW TEXT.

EXCISE - DELETE CHARACTER(S) IN A RANGE OF LINES. BEGINNING WITH SPECIFIED CHARACTER POSITION, DELETES DESIRED NUMBER OF CHARACTERS.

SEARCH - SEARCH FOR STRING IN RANGE OF LINES BELOW THE CURRENT LINE - LINES THAT HAVE THE STRING ARE PRINTED. IF NO. OF LINES IS NEGATIVE, SEARCH ENDS WHEN FIRST FOUND. (DEFAULT=-4999)

CHANGE - REPLACE STRING IN A RANGE OF LINES - LINES THAT HAVE BEEN MODIFIED ARE PRINTED. IF NO. OF LINES IS NEGATIVE, SEARCH ENDS WHEN FIRST FOUND. (DEFAULT=-1)

REPLACE - REPLACE THE CURRENT LINE WITH NEW TEXT  
IF CR IS GIVEN INSTEAD OF TEXT, NO CHANGE IS MADE

FILE - WRITE FILE - SAVES CURRENT CONTENT. EDITING IS NOT TERMINATED. EXAMPLE: 'FIL NEWTXT.RNO'

QUIT - QUIT EDITING - THIS COMMAND DOES NOT WRITE A FILE

? - PRINTS THESE INSTRUCTIONS

IF YOU ABORT THE PROGRAM WITH A 'CONTROL-C', A 'GOTO 1670' COMMAND CAUSES EDITING OF THE EXISTING TEXT TO BE RESUMED.

NOTE: NORMALLY YOU NEED NOT BE CONCERNED WITH THE FOLLOWING - THE WORKSPACE IS A 'VIRTUAL CORE' FILE NAMED 'EDIT.TMP'; IT IS AUTOMATICALLY DELETED BY THE 'QUIT' COMMAND.

### OLYMPIA AREA COMPUTER CLUB

We have had a club in Olympia for over a year, known as Olympia Area Computer Club (OACC). It has now come out of the woodwork and meets each third Wednesday of the month at the Olympia Public Library (until further notice).

No dues or officers yet or planned. We do have some special interest groups, so come and see - 7 PM, 3rd Wednesday. For info contact Scott at 943 - 1938.

### FEEDBACK

In your February newsletter you discussed the "slow-down," Mits, etc., and I am moved to comment. We all have been made aware that the hobbyist industry is showing finite limits. Hardware companies and stores all around are emphasizing business users, rather than hobbyists—that much is necessary and is OK with me. But so many have acted as if hobbyists are bad for business and not quite nice. If it weren't for us, maybe business micros wouldn't be off the ground today. There must have been a lot of arguments in the 1930s about hams being a big waste of the radio spectrum, which business was just finding uses for. This kind of attitude is not necessary, and it's not good business.

Amateur computerists, as a rule, make their living some-

where in the business world—and every business is a user of information processing. Many Northwest Computer Society members use computing in some way in their jobs. Our co-workers respect our knowledge of this new thing, the microcomputer. When others at work asked me for help picking out a microcomputer system for use on the job, did I take the guy with the bucks into the stores that didn't want to talk any more to a hobbyist? No.

I don't think a storekeeper has to hurt himself or to waste large resources on serving hobbyists. I think he should consider that the majority of hobbyists have some positive influence on business sales.

**John P. Aurelius**  
Secretary  
Northwest Computer Society  
PO Box 4193  
Seattle WA 98104



TRS-80 PRIVACY - CRYPTTEXT is a hardware encryption device designed to plug directly into the back of the TRS-80 or into the expansion interface via optional cable. It can be used to store encrypted info. on cassettes or disks, or to transmit via modem. To decode data, one must have: the CRYPTTEXT unit, software, and the correct key (10 characters). Throughput is greater than 15K bytes/sec. Under \$300, includes demo software and documentation. Details from CRYPTTEXT Corp., P O Box 425, Seattle, WA 98125, (206) 364 - 8585

BLOWER - 12VDC Axial Blower, high volume air over motor direction, \$1.50 ea. A sample of this item was received and shown at the NWCS meeting on 7 June - looks like a nice small fan to blow onto a hot - spot in your hardware. American Surplus Trading Co., 332 Canal St., New York, NY 10013, (212) 966 - 5650.

BOOK - "Systems Extensions" is a 124 page book in 8.5 x 11 inch soft cover format. The first 60 pages are editorial content with articles such as: Computers of the Past, -Present, -Future; Methods to Program Your System; Software Background Report; The TRS-80 and the Business Community; etc. The remainder is an informative catalog describing software, supplies and furniture. While much of the information is general in nature, the slant is definitely towards the TRS-80. "Systems Extensions" is \$3 from The Bottom Shelf, Inc., P O Box 49104, Atlanta, GA 30359, (404) 938 - 3304.

DISKS - Verbatim, 3M, all types and sizes. Plastic pages and library cases for disk storage. Example prices: Verbatim 8" soft sectored, List \$5.45, 10-40 \$3.50, 50-190 \$2.90. 3M 5" soft sectored, List \$6.50, 10-40 \$4.20, 50-190 \$3.50. Data sheets from ComputeSystems Distributors, Inc., 3470 Erie Blvd. E, Syracuse, NY 13214, (315) 446 - 1285 and (800) 448 - 5523.

DISCOUNTS - Our plan is to offer club members discounts on items as: the GSI-SIEMANS FDD 120-8 8" floppy disk drive in group purchases for \$375; Tarbell controllers assm. \$219 with purchase of drive; Cherry Pro keyboard at \$99; 12" used video monitors (new tube) \$99. In stock 16k memory 300 ns for S-100 \$365, 32K \$580. Catalog. Electrolabs, P O Box 6721, Stanford, CA 94305, (415) 321 - 5601.

BUY AND SELL - The Lectronics Emporium is a classified news sheet with ads for amateur radio and computer equipment. Also test equipment and components. Monthly, \$3/yr. Dick Costello WB1ABU, Pub., P O Box 828, Derry, NH 03038, (603) 434 - 1744.

HELP! We wish to build a microcomputer as much from scratch as possible and we need to find some "how to design and build" information or some schematics that we can work from. Do you have any idea if such cookbook - type things are available? ANY help you could give would be greatly appreciated. Betty Irwin, Hobbett Audio, 3501 Commercial, Vancouver, BC V5N 4E8, Canada.

UPGRADE KITS - For Apple II and Sorcerer; Ithaca Audio expands its line of high density, high quality 16K memory expansion kits with the introduction of two more Simple Up-Grade Kits. \$140, in stock at Magnolia Microsystems, 3214A W McGraw St. 7, Seattle, WA 98199, 285 - 7266.

STOCK ANALYSIS - I wish to make contact with others on the subject of trend analysis in commodities and stocks. I have years of successful experience and wish to exchange ideas, etc. Ed Tenberg, Box 407, Little Neck, NY 11363.

TRS-80 SOFTWARE - COTS, the Clearinghouse of TRS-80 Software, is a Canadian firm trying to simplify the buying and selling of TRS-80 compatible software and hardware across the border. A newsletter is published at \$12 for 12 issues, articles, classifieds, listings of COTS software and hardware. Software for distribution and articles for the NL needed. contact: COTS, L E Whalen, P O Box 3103, Ottawa, ON K1P 6H7.

TRS-80 PASCAL - An outfit in Australia is offering a tiny Pascal for the TRS-80! Runs on a 16k LEVEL II single cassette machine, a sub-set compiler based on "BYTE Magazine" Chung/Yuen articles. Editor/compiler, pcode interpreter, pcode to z80 translator, runtime system, tiny Pascal library, sample program, listings, documentation, instructions, run time system source on cassette, airmail postage... all \$60 US. John Alexander, Pipe Dream Software, 28 Palmerston St., Berwick, VIC.3806, Australia. You can send for a data sheet, or the very rich can call: (03) 707 2851.

BUY AND SELL - Bits and Pieces is a monthly classified newsletter. Intro offer - free 3 line ad, free copy. P O Box 36, Commack, NY 11725.

WANTED - Motorola Evaluation Kit II in any condition for parts. Charles Worstell, 36012 Military Rd. S, Auburn, WA 98002, 927 - 6038 (Tacoma).

SELLING last of S-100 BOARDS - Imsai PIO4-1 assembled, tested, new, \$80; also Demo and used: Vector Graphic PROM+RAM, \$75; Poly Idea Board, \$50; Extendsys 32K DRAM Memory, \$400; Imsai proto boards, \$20; MITS 4PIO-1 board, \$75; MITS 1KRAM, \$40; TCH Proto board, \$15; PROM setter, \$100. Bernie Brunson, 488-4441 (Bothell).

TELETYPE MODEL ASR-33 WITH AUTO-PUNCH \$400 ... 226-2880 ... RON COWLES

SOFTWARE - KISS is a multi keyed indexed sequential search file control, \$535; user guide alone \$25. K-BASIC Microsoft Disk Extended BASIC with KISS DBMS \$995. For ISIS-II and CP/M operating systems. Data sheets. Eidos Systems Corp., 315 Wilhagan Rd., Nashville, TN 37217, (615) 242 - 8893.

WANTED - SOL-20 non working for cabinet or working if reasonable. Bob Schaeffer, P O Box 4-1983, Anchorage, AK 99509, (907) 344 - 0082 evenings (Pacific time + 2 hrs).

FOR SALE - Okidata CP110 Printers, 110 cps, bidirectional matrix uppercase, RS232 interface, tractor feed. Working and tested (4) \$450. Prints, but bad RS232 (4) \$300. Not working (3) \$175. 745 - 0162 (Seattle) eves., 784 - 5482.

COMPUTERS - TEI offers a wide range of computers, including Processor Terminals (8 models), Terminal Systems (6 models), S100 Mainframes - shielded, grounded and actively terminated - (8 models), CPU Boards (8080A or Z80), RAM Board, I/O Board, Floppy Disc Controller Board, Video Board 24 x 80, Shugart Drives, Drive Chassis. CP/M DOS, Super-BASIC, Fortran IV. Catalog. TEI Inc., 5636 Etheridge, Houston, TX 77087, (713) 645 - 4821.

TARBELL - Products include: 32k RAM Memory \$625, 16k RAM Memory \$390 (assembled, 300ns, static, bank switching, low power). Cassette Interface, the industry standard, up to 540 bytes per second \$120 kit or \$175 assm. Prototype board, \$28. Cassette and disk BASIC, 24k, available on cassette and CP/M disk, \$48 each; source on paper or disk \$25 extra. Ptech. Software Pkg 1, \$5 ea., cassette listing. EMPL, a micro APL for 8080 in 5376 bytes, cassette \$15, CP/M disk \$20. CBASIC-2 compiler for CP/M, diskette and manual, \$100. Catalog. Tarbell Electronics, 950 Dovlen Place, Suite B, Carson, CA 90746, (213) 538 - 2254.

LIQUIDATION - Computer portrait equipment: Exidy Sorcerers (2 ea.) 8K \$789, 16K \$945. Integrand chassis (1) \$290. Integral Data printers (2) \$885, Malibu Model 160 line printers (3) new \$1195; (1) demo \$1095. RCA TC1000 Video Camera \$185 w/o lens, lenses \$15-165. Panasonic 9 inch monitor \$165. Portrait systems with digitizing equipt., computer, line printer, camera, lens, monitor, cabling, software, heat transfer ribbons, heat press - the works: under \$3000. We're not quitting, going into color. Details from Haggard Kristen, Inc., 4535 S Padre Is. Dr. 18, Corpus Christi, TX 78411, (512) 855 - 3857

FORTRAN STRINGS - The String Bit is a character string handler for FORTRAN and written in that language. \$45 for source code, demonstration program, users manual and 5" or 8" CP/M compatible disk. Free data sheet. Key Bits inc. P O Box 592293, Miami, FL 33159.

DISCOUNTS - Sorcerer, Hitachi Monitors, Hazeltine 1400, Tarbell Disc Ctrlr., TI Calculators, etc. Catalog 50 cents. Hollywood Systems, 9100 Sunset Blvd Suite 112, L A, CA, 90069.

HEATH H-8 SYSTEM - Computer, serial and parallel I/O, 24K, H-9 CRT, TTY Printer, tape recorder, desk, \$2300. H-17 Dual disk in warranty \$900 - no problems. Both \$3000 (over \$3500 new). Tom McKenna, 12428 - 68th Ave. NE, Kirkland, WA 98033, 823 - 5880.

NORTHSTAR SYSTEM - IMSAI 8080N with Northstar operating system, keyboard, video display and 16K memory. \$2000. Mike O'Quinn 226 - 0493 after 5 pm.

FOR SALE - IMSAI 16K dynamic memory board. Works fine - except with my DMA floppy disk controller. Gale Sherry, 783 - 0853.

CLASSIFIED ADVERTISING: The Buy and Sell Forum for the Computer Hobbyist is "ON-LINE", mailed first class every third Wednesday. Subscriptions: 18 issues (approx. one year) \$7. Dave Beetle Publisher, 24695 Santa Cruz Hwy., Los Gatos CA 95030.

RAM FOR SALE - (4) 16K Dynabyte RAM, factory assembled and tested, \$250 ea. John G White, 216 E 5th St., Port Angeles, WA 98362, 457 - 3917.

HELP - Anyone with any experience with ELF II kit featured by RCA and mentioned in 1979 issues of Pop. Electronics. Contact Oliver 455 - 4266 or 455 - 5833.

DISKS - Verbatim Mini - Disks at low cost in quantity. 500 disks \$2.50 ea., 1K disks \$2.25 ea. Mix or match Verbatim 525-01, -10 and -16. Data sheet. Disks, Etc., P O Box 327, Center Valley, PA 18034

FOR SALE - S-100 Buss System. Cromemco Z-80 CPU, BYTE-8 Mainframe, TDL System Monitor board, ACT-I Keyboard, North Star disk and software, 2 - 16K static ram (250 nsec, less 8k of chips), Panasonic video monitor. Up, tested and running. For more details send SASE or call (206) 456 - 2466 after 5 pm. Donald A. Coulter, 8002 Mountain - Aire Loop SE, Olympia, WA 98503.

BUSINESS SOFTWARE - Micro Mike's Program Library provides members access to a wide variety of high quality programs at a very reasonable cost. You join by: paying \$500, or donating a program (if accepted), or purchasing \$2500 in hardware from Micro Mike's. A large selection of hardware is carried, including North Star. Free data. Micro Mike's, Inc., 905 S Buchanan, Amarillo, TX 79101, (806) 372 - 3633.

BOOK - How to Package and Market your Own Software Product and Make it Go - 185 pp in ring binder \$45, 30 day exam. Free data sheet. Datasearch, Inc., 4954 William Arnold Rd., Memphis, TN 38117 (901) 761 - 9090.



## RANDOM NUMBERS

by Joe Felsenstein

Well, it's Back to BASIC again. This time I want to talk about a searching technique. It's rather elementary. Sophisticated types who have read everything published in all the microcomputer mags and do a lot of machine language programs can use this column to wrap their Fish 'n Chips in, but maybe there are one or two people out there who will find this useful.

### BINARY SEARCHES

What I want to do is to describe a common trick for searching files, and give a BASIC fragment to do it. The technique is called Binary Search. Suppose that you have an array of numbers, or a file containing numbers, and you simply want to know where in the array (or file) a particular number is. The obvious and simple-minded way is simply to write a program which starts at one end of the array (file) and looks at one element (record) at a time. When it finds the number it wants, it prints out some indication of where it is, then stops. On the average, you will go half way through a file before finding the one you want, if you are looking for a randomly chosen entry. So to look for entries in an array (file) of 5000 elements (records) you have to look at 2,500 entries.

When you do things this way, you are in effect assuming that you don't know anything about the order of the entries in the array. But suppose that you do. In particular, suppose that the entries are in order. There are many applications for which they will be, such as an address file in which your friends (customers, victims) are listed alphabetically. This enables a great saving, as follows.

Suppose that you know that the array is 5000 entries long, and that the entries are ordered in increasing order, with the smallest entry first. Suppose that you started not by looking at the entry A(1), but at A(2500), the entry in the middle. If it is less than the number you seek, then you know that the one you want is in the last half of the array. If it is greater than the one you seek, you know that that one is in the first half of the file. Now all you need to do is to narrow things down still further.

Obviously, you can just keep up with the process of looking in the middle of a stretch of numbers. For example, if it turned out that you number is somewhere in the stretch from A(1) to A(2500), you look next at the middle of that stretch, say at A(1250). You keep up this process. At first you have 5000 numbers to look at. But after one examination, you have only 2,500, then 1,250, then 625, then 313, and so on.

Only a few examinations are needed to find the element we want. In fact, the number is about the same as the number of times we have to halve 5000 to get it down to 1. This is the logarithm to the base 2 of 5000, which is between 12 and 13. You can see that this is a much better way to do things. Instead of 2,500 examinations of numbers, we need only 13 or so. For a large array (or a long file) things are even better. If there were a million entries, you would need 20 examinations instead of half a million!

It's not difficult to write a program to do this kind of search. In the simple case of looking for numbers in an array, here's all you need to do:

```
100 LET L=1
110 LET U=N
120 IF L=U THEN 190
130 LET I=(L+U)/2
140 IF A(I)>=X THEN 170
150 LET L=I+1
160 GOTO 120
170 LET U=I
180 GOTO 120
190 REM EXIT HERE
```

This program fragment is fairly simple. The array A(1) through A(N) contains the number X somewhere, and we want to find where. At the end, A(L) will contain either X or a number very close to it if X isn't anywhere in the array. Of course, you could then test to see if you had found the number by asking IF A(L)=X. A version of this program to look for entries in a file would be a bit more complex, but only because of the necessity to go to the I-th record in the file and read part of it. The bookkeeping of the actual searching would otherwise be identical.

An ordinary file in a microcomputer is a sequential file, that is, it can only be read by starting from one end and proceeding to the other. If the file is on cassette tape, you're stuck with sequential access of the file, and you can't just go read the middle record in the file. But if the file is on disk, in most systems it is possible to do Random Access (it is in disk versions of many BASICs). This is perhaps better described as Direct Access, because you go directly to (say) the 2500th record and examine it, without reading everything in between. You don't actually read a randomly chosen record, even when it's called Random Access!

You can see how much saving can be involved (in terms of number of times the program has to read a file) by having some device like disks rather than having files stored on cassettes.

Binary searching sounds great, and it is, but amazingly enough it is not the best you can do. If you want to find out about methods of searching files (or arrays), there is a wonderful source in an article by Donald Knuth (he of the massive books on programming) in the April, 1977 issue of Scientific American. The article is entitled "Algorithms", and Knuth uses this subject to exemplify the ways in which unexpected savings can be achieved. He tells about a tricky technique called hashing, which can take as little as one-and-one-half examinations to find an entry! But binary search is a lot easier to program, and if you think about it, the drop from 2,500 to 13 (a factor of 200 is more significant than the drop from 13 to 2, especially if you consider the amount of computer time involved).

The lesson is a simple one: if you know things about the way your data is arranged, make use of them if possible. Here the knowledge that the data was arranged in increasing order saved a factor of 200 in the number of times the array had to be looked at. I leave it to you to figure out how to do things if the array is in decreasing order. P. S.: this program HAS been debugged, believe it or not (but of course errors are not impossible).

### RUMOR

From several (well, two) sources I have heard that Processor Technology has gone belly-up. This is not unexpected, as we are entering a shakedown period in the microcomputer market, with the biggies moving in (TI will be announcing very soon, supposedly, and I have heard it said

that Hewlett-Packard will be out with a microcomputer in the \$500 range which, amazingly enough, will have a liquid crystal display rather than one you plug into your TV).

Small outfits like (the late) Proc Tech, or (the late) IMSAI, or (the late) UmTech (= VideoBrain) always seem to have the same problems: inability to find capital, and inability to market on a large enough scale. Only Apple of all the small outfits seems to be doing well in getting its product out. A year from now the whole microcomputer market will look much different. Will the computer store itself survive? I find it hard to believe that outfits like TI and HP will distribute through computer stores, at least not the existing ones.

In case some lawyer is tempted to write letters and complain, these are partly just rumors (though in some cases fairly authoritative ones), and I'm not really sure which outfits are in bankruptcy and which in reorganization (from which it is theoretically possible to emerge alive). There are other casualties littering the field of battle too, but those are the ones which spring to mind.

Of course, by the time this comes out you may have heard all this and more. Rumors travel faster than newsletters.

### AMAZING COINCIDENCES

I have been more and more impressed by the way that DEC's way of doing things has become the microcomputer standard. Many products on the market are very much like DEC (I dare not say more than that). For instance, Microsoft BASIC is similar to DEC's BASIC-PLUS (or so I hear), and the most popular 8080 disk operating system, CP/M, is very close in commands to a simplified version of DEC's TOPS-10 operating system for the DECsystem-10 (maybe it is even closer to some PDP-11 operating system I don't know).

In CP/M, if you want to make a copy of one file (called CRUD.HUH) and call that copy GARBGE.OK, you simply type:

```
PIP GARBGE.OK=CRUD.HUH
and the copy happens. By comparison,
in TOPS-10 you type
COPY GARBGE.OK=CRUD.HUH
which is virtually identical. In
fact, TOPS-10 used to use (and some
PDP-11s still use) the word PIP
instead of COPY!
```

If you're not impressed by the similarity, here's what you have to do on the CDC 6000 series operating system NOS/BE:

```
COPYE(CRUDHUH,GARBGE)
which isn't very similar, though maybe
you can see what it is doing just as
easily.
```

To give you an idea why people like to do things DEC's way, here is what I had to do to submit a job (in fact, to compile a FORTRAN program and run it) on the University of Chicago's remarkably inadequate IBM 360/50:

```
//CLUMP JOB (acctnr,junk,more),
'passwd',TI=(6),REGION=250K,LL=
20000,
//Q=0
// EXEC FORTGG
//GO SYSLOBJ DD DSN=junk.more.more,
UNIT=SYSDA,VOL=SER=STOR04,
// DISP=SHR
//GO.SYSIN DD *
```

I have indented because the cards were so long, and rendered all the secret passwords in lower case (each card starts with two slashes). To be fair to IBM, this isn't the simplest possible FORTRAN compilation (I think it gets a file off disk and tells the system that it is the one to use). If it seems incomprehensible to you, it's because IT IS. There are alleged to be those who understand this (it is called JCL - Job Control Language). 'Nuf said?

\* SYSTEMS ANALYSIS & DESIGN, WITH AN EMPHASIS ON SYSTEMS FOR SMALL BUSINESSES.

### Computer Resources

Data Processing Consultant  
ROY S. GILLETTE

(206) 523-2866



```

010 REM DOUG KEITHLEY
020 PRINT "THIS PROGRAM WILL CONVERT RESISTOR COLOR CODES TO RESISTANCE"
030 PRINT "VALUES. IT ALSO SOLVES SIMPLE SERIES OR PARALLEL CIRCUITS"
040 PRINT "GIVEN APPLIED VOLTAGE AND RESISTANCE OR RESISTOR COLOR."
080 FOR A=0 TO 9
090 READ B$(A)
100 IF A>3 THEN 130
110 READ C$(A)
120 READ D(A)
130 NEXT A
140 PRINT"WHICH OF THE FOLLOWING DO YOU WANT TO DO:"
150 PRINTTAB(13); "1) CONVERT COLOR TO RESISTANCE OR VICE VERSA."
160 PRINTTAB(13); "2) COMPUTE SIMPLE SERIES CIRCUITS."
170 PRINTTAB(13); "3) COMPUTE SIMPLE PARALLEL CIRCUITS."
180 INPUT "PRESS THE NUMBER OF YOUR SELECTION";E
190 IF E=1 THEN 220
200 IF E=2 THEN 220
210 IF E>3 THEN 180
220 PRINT"PLEASE ENTER THE RESISTANCE AND % TOLERANCE(SEPARATED"
230 PRINT"BY A COMMA). OR, TO ENTER THE COLOR CODE, ENTER '1,1',THEN"
240 PRINT"ENTER THE FOUR COLORS SEPARATED BY COMMAS."
250 PRINT"THE FOLLOWING ARE THE RECOGNIZED COLORS:"
270 FOR A=0 TO 9
280 PRINT B$(A);", ";
290 NEXT A
300 PRINT"NONE"
310 PRINT"TO END THE INPUT OF RESISTANCE VALUES"
320 PRINT"TYPE '0,0' "
370 FOR A=0 TO 9
380 PRINT"RESISTOR #";A;
390 INPUT F(A),G(A)
400 IF F(A)=0 THEN 690
410 IF F(A)=1 THEN 530
420 FOR H=0 TO 9
430 IF INT(F(A)/10↑L)=0 THEN 450
440 NEXT H
450 I$(A)=B$(INT(F(A)/10↑(L-1)))
460 J$(A)=B$(INT(F(A)/10↑(H-2))-INT(F(A)/10↑(L-1))*10)
470 K$(A)=B$(H-2)
480 FOR H=0 TO 3
490 IF G(A)=D(H) THEN 510
500 NEXT H
510 L$(A)=C$(H)
520 GOTO 680
530 PRINT" COLORS ";
540 INPUT I$(A),J$(A),K$(A),L$(A)
550 FOR H=0 TO 9
560 IF I$(A)↔B$(H) THEN 580
570 M=L
580 IF J$(A)↔B$(H) THEN 600
590 N=L
600 IF K$(A)↔B$(H) THEN 620
610 O=L
620 NEXT H
630 F(A)=(M*10+N)*10↑O
640 FOR H=0 TO 3
650 IF L$(A)=C$(A) THEN 670
660 NEXT H
670 G(A)=D(H)
680 NEXT A
690 FOR A=0 TO 9
700 PRINT"RESISTOR #";A,F(A);"OHMS, ±";G(A);"% ";
710 PRINT I$(A);J$(A);K$(A);L$(A)
720 NEXT A
730 IF E=1 THEN 920
740 FOR A=0 TO 9
750 IF F(A)=0 THEN 810
760 IF E=2 THEN 790
770 M=1/F(A)+M
780 GOTO 800
790 M=F(A)+M
800 NEXT A
810 IF E=2 THEN 830
820 M=1/M
830 PRINT"WHAT IS THE APPLIED VOLTAGE?";
840 INPUT N
850 PRINT"TOTAL RESISTANCE IS ";M;" OHMS"
860 PRINT"RESISTOR-RESISTANCE-CURRENT-VOLTAGE-POWER"
870 FOR A=0 TO 9
880 PRINT A;TAB(9);F(A);
890 IF E=3 THEN 920
900 PRINTTAB(20);N/M;TAB(28);N/M*F(A);TAB(36);N↑2/M↑2*F(A)
910 GOTO 930
920 PRINTTAB(20);F(A)/N;TAB(28);M;TAB(36);M*F(A)/N
930 IF F(A+1)=0 THEN 940
935 NEXT A
940 DATA "BLACK","RED",2,"BROWN","GOLD",5,"RED"
950 DATA "SILVER",10,"ORANGE","NONE",20,"YELLOW"
960 DATA "GREEN","BLUE","PURPLE","GREY","WHITE"
970 END

```

--gets  
--color code

--gets next A

--'oh equals L'

--read last expression  
in this 'ten to the  
oh power'

**THE TRS-80 USERS' GROUP**  
by Swend Miller

The TRS-80 users' group meets on the second and fourth Wednesday of each month. The meetings are presently held in the recreation room at the Pine Terrace trailer village, 21814 Pacific Highway South in Des Moines. The Pine Terrace is approximately one block south of the intersection of 216th St. and Pacific Highway South and is on the East side of Pacific Highway South. When parking, don't block anyone's driveway. Ask at the meeting if you're not sure. Our meetings are usually informal (ties are not required) and visitors of all ages are welcome. We encourage owners of systems other than TRS-80's to bring them for demonstration and idea exchange. There are usually several TRS-80 systems operating at each meeting. At future meetings our group will be conducting a forum on disk system file management and a class on structured programming in the planning stages. The following people may be contacted for information concerning the group and its activities:

Dick Keller Phone: 762-4459  
Walt Nash Phone: 824-4063  
Swend Miller Phone: 631-5694

We hope to contribute on a regular basis to the Northwest Computer News, so if you have information applicable to the TRS-80 group, please contact us.

**CP/M USER'S GROUP**

**Dick Curtiss 784-8018**

Until further notice meetings will be held the last Wednesday of each month except December when there will be no meeting. Time: 7 PM to 9 PM.

KOMO  
Studio G  
4th N. & Denny Way  
Seattle

Northwest Computer News accepts limited, relevant commercial advertising. Ads will reduce our cost to produce the News and, we hope, will keep us informed of new products, services and opportunities as they appear on the marketplace. Current rates as of May, 1979, follow:

size	insertions	1-2	3-5	6-12
Full page		\$90	\$85	\$80
2/3		60	55	50
1/2		50	45	40
1/3		35	30	25
1/4		30	25	20
1/6		20	16	12
1/10		15	12	9
1/15		10	8	6

— prices above per insertion —

Special — Six insertions of invariant copy 1/30 page (business card) \$12 total.

Camera-ready copy assumed. Arrangements made for special work at moderate cost. Terms are cash with order. Net 10 days from invoice to well-rated firms. Rates are subject to change without notice.

reprinted from the Inland Empire Computer

Club Micronotes, June, 1979.



reprinted from the Inland Empire  
Computer Club Micronotes, June, 1979.

## Venturing Forth

by Ron Hodges

At the dawn of the micro-computer era, I ran across mention of a language called "FORTH." I had experience with several higher level languages and a few assembler or machine level languages, and I was always looking for the new and different subjects in the field of computers. I wrote and asked for information on this language, and eventually a small package arrived that did more to confuse than to enlighten.

As far as I could tell, "FORTH" was a totally different type of language, which had all the advantages of both an interpreter and a compiler, with none of the bad points. Typical sales pitch! The bottom line, however, was the price. Only \$2500 for a microprocessor based installation. And that system must have a disk I/O capability. It was obvious that neither my 16k tape-based system nor my bank account would accept it. I let the matter rest until a couple of months ago.

At that time, I saw an ad by Programma International, Inc., which offered a version of "FORTH" for the 6800. I called them and found that this version only needed 12k of memory and would work on a tape-only system. I sent them the \$35 they asked and embarked on an enjoyable journey.

About the only way to describe "FORTH" is to say that it is strange, but pleasantly so. Probably the easiest way to give an idea of how it works is to compare it to a dictionary. In fact, a large part of the "FORTH" program is just that. Imagine buying a very simple dictionary, one that only contained a very few words. These words had been carefully chosen, however, so that they were all you needed to define new words. In "FORTH," the dictionary contains the basic words or operations needed to perform the standard type of log-

ical and arithmetic manipulations common to computing problems. Since the words can be composed of symbols as well as letters, even the common mathematical operations (+, -, \*, etc.) exist as "words."

In use, "FORTH" accepts input lines from the keyboard just as "BASIC" does. After you have typed in the line, the program scans the line looking for the individual words or symbols. You must separate each entry from the others with at least one space. As each word is found, the dictionary is searched to see if the word is present. If so, the machine level code is executed and control is passed to the scan routine. If the word is not found, an attempt is made to evaluate it as a number in whatever base you are working in. ("FORTH" can operate in decimal, binary, and octal.) If this can be done, the number is placed on a 'stack' for use later. This stack is not the processor stack common to all microprocessors, but an internally managed logical stack of "FORTH." This stack works just like the stack on the popular Hewlett-Packard pocket calculators in that operations are performed in "Reverse Polish," or RPN, notation. To add two numbers, they would be placed onto the stack, the "+" operator would be given, and then the print command would be used to show the results. In "FORTH," the print command is the dot, ".". The commands to add 258 and 319, print the result, and leave the answer on the stack would be:

```
258 319 + DUP .
```

In operation, the two values go onto the stack as they are entered. The "+" operator adds the two values together, removes both from the stack, and places the result back on the stack. The "DUP" will duplicate the last stack entry, leaving two copies. Finally, the "." operator removes the latest stack entry and prints it out. Due to the "DUP" function, one copy of the result is still on the stack for later use.

There are many of these "primitive" operators in the usual versions of "FORTH." In addition to words for manipulating the stack and its contents, there are words to retrieve and store data, and for setting up new word definitions. This facil-

ity for allowing the programmer to define new words is what makes "FORTH" so attractive.

To define a new word, the ":" operator is used, followed by the word you wish to define. Following this word, any words or values specified will not be executed immediately. Instead, machine code to call these routines will be placed in the dictionary entry for the new word. The definition is then ended with the ";" operator. For example, to define a word "Function" to evaluate the expression "3X + 9", the following could be used:

```
: FUNCTION 3 * 9 + ;
```

In use, a number would be placed on the stack, "Function" specified, and then "." used to print the result. If desired, other words could be designed to use this new word. For example, suppose there were two values on the stack. Each is to be evaluated according to "Function" and the difference found. We could define a new word, "DIFF", to be:

```
: DIFF FUNCTION SWAP  
FUNCTION - ;
```

The first "Function" evaluates the top number on the stack, replacing it with the result. Then "SWAP" is called to exchange the top two stack entries. This places the other original value at the top of the stack for "Function" to evaluate it. Then "-" is called to subtract the two items. The result is left on the stack. To show the use of this, look at the example:

```
513 DIFF . (Result is 24)
```

Remember in trying to work out this example that since 13 is the latest item on the stack, it will be evaluated first!

It can be seen where the term "threaded" language is obtained. Each definition contains "threads" linking it to other definitions, each of which in turn contains still more "threads" to still more definitions. This can continue to any depth. Since these "threads" are merely subroutine calls at the machine code level, "FORTH" programs run very fast. Usually, this is only slightly slower than hand-coded machine language. It is certainly 100 to 1000 times faster than BASIC! As opposed to BASIC, where the

actual ASCII text is stored as the program, in "FORTH" the program consists of dictionary entries for each word, followed by actual machine language code to execute these words. By using a linked list for the dictionary, searching for a word is very fast. After the word is found, the code execute is actual machine code, not some intermediate or interpretive text.

In keeping with current trends in programming, it should be pointed out that "FORTH" is totally structured. There is no "GO TO" type statement. There are provisions for looping and conditional structures similar to the familiar "FOR-TEXT" and "IF-THEN-ELSE" constructs in BASIC. However, they cannot extend outside the range of a single word definition. This makes testing a simple job. Either the new word works or it doesn't. If it doesn't, you don't have to go looking all over the program to find the error.

The testing of "FORTH" routines is extremely simple. As each new word is defined, it can be tested by supplying on the stack the values it expects, executing the new word, then looking at the stack to see if the result is correct. Once tested, the word can be used in other definitions with the assurance it will function as desired. If this testing is done as the program is written and entered, it is almost sure to "work the first time"!

It should be obvious that this is merely a brief introduction to "FORTH." It does take a bit of getting used to in order to write usable programs. However, that task is very pleasant and the result is certain to be rewarding. At this point I cannot recommend that you purchase the 6800 version of FORTH from Programma International, Inc. While most of it works as advertised, the editing functions do not, and this makes program development quite a task. They do offer versions for the PET, TRS-80 and the Apple, but I have no experience with these. For those confirmed 6800 fans out there, if you can wait a few weeks, I will have my own versions of "FORTH68" up and running. I promise a demonstration at a meeting if there is enough interest. I plan to market this package and hope to have it ready by September.

## Pascal for Northstar Disk Users

by Ernie Kent

I have gotten the Pascal package from Northstar up and running. Since it seems nobody else has yet received it, here are a few pointers to save you time and grief when it comes.

The only problems with the documentation are that the procedure for BINDING the GOTOXY procedure is on a missing page of the manual, there is a misprint and some confusion in the description of the setup options, there is an unmentioned problem with doing the BINDER on single density, there is a danger in using K(runch while working on your I/O primitives, and some confusion about the N\* prom address.

1. The documentation and the disk directories all refer to a starting prom address of E800. Maybe that's something new since I got my disks? Anyway, it all works if you ignore it and use E900 as usual (at least as per my usual.)

2. Always use the N\* CD command to copy your disks. It can be done using the Pascal facilities plus the Booter routine (for the N\* disk I/O blocks that don't appear in Pascal's files), but CD is quick and safe.

3. If you want to use the system beginning with RAM at 0, be sure to make an I/O routine appropriately relocated for both 0 and 2000, and patch both USERIO.2 and UERIO.0 as described in the addendum, bring up the disk (which will come up on SYSTEM.NSTAR2), and then R(emove SYSTEM.NSTAR2. Next time the disk comes up it will use SYSTEM.NSTAR0. If you want to start at 2000, R(emove SYSTEM.NSTAR0. So far so good, and all as described in the manual addendum. However, if you are going to use an X/Y addressable screen,

you're going to have to use the BINDER routine. The BINDER routine requires 60 (512 word) blocks of free space, and there is no such lump on the single density disks at this point. You must make the room. One way is to delete the files that the BINDER doesn't need and then K(runch the disk. However, if you do this,

SYSTEM.NSTAR gets relocated and the USEP I/O blocks are now "somewhere" other than where the N\* directory says they are. If you now try to remodel your I/O routine after testing your new GOTOXY procedure and put the remodeled version in the same place as before, the system goes into convulsions and dies. There is a way to find the lost I/O blocks. Before doing the K(runch, use the E(xtended directory command of the F(iler to find where SYSTEM.NSTAR resides. Do this again after the K(runch. By noting where the I/O blocks were in the N\* directory, and by how much SYSTEM.NSTAR was moved in the K(runch it is possible to calculate the new location of the I/O blocks if you bear in mind that the disk addresses in the N\* directory refer to 256 word blocks, while the disk addresses in the Pascal directory refer to 512 word blocks (and begin at 10).

\*A BETTER WAY\* is to make another copy of the disk PASNS, call it PASNSX, R(emove the unwanted SYSTEM.NSTARX, R(emove all the files the BINDER doesn't need (it needs only SYSTEM.NSTAR, SYSTEM.PASCAL, SYSTEM.FILER, SYSTEM.MISCINFO, and BINDER.CODE). Then K(runch the disk, X(ecute the BINDER, bring up the unK(runched version of PASNS, and transfer the SYSTEM.PASCAL from the disk altered by the BINDER (PASNSX) to the unaltered disk (PASNS). Now you can remodel your I/O at the location shown in the N\* directory after testing it with the modified SYSTEM.PASCAL containing your new GOTOXY. Later on when it's all just right, you can remove the BINDER and other unneeded files from this disk and K(runch it.

4. When using the SETUP.CODE routine to customize your SYSTEM.MISCINFO, note the misprint at the bottom of p. 225 of the manual. The manual says "Move cursor left" one of you can teach me how to use the program says "move cursor right". The language itself! We should also give some program is correct. Note also here that in thought to how we are to access the the manual these options are grouped according to whether they refer to keyboard available on the large disks.

codes, instructions to the video drivers, etc., while as they come out of the SETUP program, they are alphabetically arranged. It is easy to get confused about which is which.

5. Good Luck. When the rest of you N\* types receive your Pascal packages, maybe one of you can teach me how to use the language itself! We should also give some thought to how we are to access the library of Pascal programs that are available on the large disks.

