

For superior OS-9 performance, the  
**SYSTEM V**

Provides a 68020 running at 25 MHz, up to 128 MBytes of 0 wait-state memory, SCSI and IDE interfaces, 4 serial and 2 parallel ports, 5 16-bit and 2 8-bit ISA slots and much more. The SYSTEM V builds on the design concepts proven in the SYSTEM IV providing maximum flexibility and inexpensive expandability.

**AN OS-9 FIRST** - the MICROPROCESSOR is mounted on a daughter board which plugs onto the motherboard. This will permit low cost upgrades in the future when even greater performance is required.

G-WINDOWS benchmark performance index for the SYSTEM V using a standard PC VGA board is 0.15 seconds faster than a 68030 running at 30 MHz with ACRTC video board (85.90 seconds vs 86.05 seconds).

Or, for less demanding requirements, the  
**SYSTEM IV**

The perfect, low cost, high-quality and high performance OS-9 computer serving customers world-wide. Designed for and accepted by industry. Ideal low-cost work-station, development platform or just plain fun machine. Powerful, flexible and expandable inexpensively. Uses a 68000 microprocessor running at 16 MHz.

Both computers provide flexible screen displays in the native mode with the optional VGA card.

**Eight text modes are supported -**

40 x 24, 80 x 25,  
80 x 50, 100 x 40,  
132 x 25, 132 x 28,  
132 x 44, 132 x 60

Foreground, background and border colors are user selectable from up to 16 colors.

**Eleven graphics modes are supported -**

640 x 200 x 16, 320 x 200 x 256,  
640 x 350 x 16, 640 x 350 x 256,  
640 x 480 x 16, 640 x 400 x 256,  
800 x 600 x 16, 640 x 480 x 256,  
1024 x 768 x 16, 800 x 600 x 256,  
1024 x 768 x 256

Text and graphics modes may be selected by a utility provided, MODESET, by software using SetStt calls or by termcap entries. In the text mode, the screen responds to standard VT100 control sequences. The full character set from Hex 20 through Hex FF is supported in text modes up to and including 100 characters wide. The upper 128 characters follow the 'IBM Character Set 2' popular with many terminals and printers. These may be displayed on the screen by using the 'Alt' key and one or two other keys (software permitting).

G-WINDOWS option provides 3 screen resolutions; 640 x 480 x 256, 800 x 600 x 256 or 1024 x 768 x 256. You can have 2 full size 80 x 25 windows with room to spare, a window as large as 122 x 44 using the large fonts or a window over 180 x 70 using the small fonts.

**delmar co**

PO Box 78 - 5238 Summit Bridge Road - Middletown, DE 19709  
402 378 2555 FAX 302 378 2556

Volume Two, Issue One

139

The "International"  
**OS9 Underground**

A **Fat Cat**® Publication

MAGAZINE DEDICATED TO OS-9/OSK USERS EVERYWHERE!

**ANNIVERSARY  
ISSUE**

- REAL WORLD OS-9
- BUILDING YOUR OWN  
C LIBRARIES
- BINARY TREES
- ANSI C
- NCODE/Dcode

# New Lower Prices! from ColorSystems

## Variations of Solitaire

Includes FIVE Variations, Pyramid, Klondike, Spider, Poker and Canfield. Complete documentation shows how to create your own games boot disk using special menu program which is included.

CoCo3 Version \$29.95

MM/1 Version \$39.95

## WPSHel

A Word Processing Oriented Point and Click Shell for all your word processing needs. Requires WindInt from your Multi-Vue Disk. Does not include editor, Formatter or Spelling Checker.

CoCo3 Only! \$20.00

We accept Personal Checks or Money Orders drawn from US Banks or International Postal Orders. NC residents please add 6% Sales Tax. Call or write for a FREE catalog!

Please add \$3 per item for shipping outside of the Continental United States.

Quality OS-9 Software for the Color Computer 3 and MM/1 from IMS

**NEW!**

## Using AWK With OS-9

A description of the AWK Programming language with an emphasis on GNU AWK for OSK. Includes the latest version of GNU AWK.

OSK Only! Just \$19.95

## OS-9 Game Pack

Includes FIVE complete games, Othello, Yahtzee, Minefield, KnightsBridge & Battleship. Includes special menu program and step by step instructions creating your own games boot disk.

CoCo3 Version \$29.95

MM/1 Version \$39.95

All CoCo3 Programs require at least 256K of memory

Coming SOON! Indexed Files for OS-9 Level 2, OS-9/68000 and OS-9000!

## ColorSystems

P.O. Box 540  
Castle Hayne, NC 28429  
(919) 675-1706

# FAT CAT PUBLICATIONS® BRINGS BACK THE PAST...

Special  
Edition

## Set of The OSKer (pronounced Oscar) Magazine

This Special Edition Set of the OSKer Magazine. News and Views in the World of OS-9/68000 and 6809, is the complete set that were published (Issues 1 thru 6). That's over 140 pages of OS-9 reading enjoyment! All pages are "comb-bound", so pages lay flat. Pages are printed to full 8 1/2" x 11" size.

Fat Cat Publications has made special arrangement with the original publisher for this reprint.

**\$19.95 (includes S&H)**

## BACK ISSUES OF THE OS9 UNDERGROUND ARE AVAILABLE...

ALL ISSUES FROM VOLUME #1 ARE AVAILABLE SEPARATELY OR IN A COMPLETE SET (12 ISSUES)

PLEASE WRITE OR CALL FOR A COMPLETE LISTING. SINGLE ISSUES ARE AVAILABLE AT THE COVER PRICE (S&H INCLUDED IN PRICE).

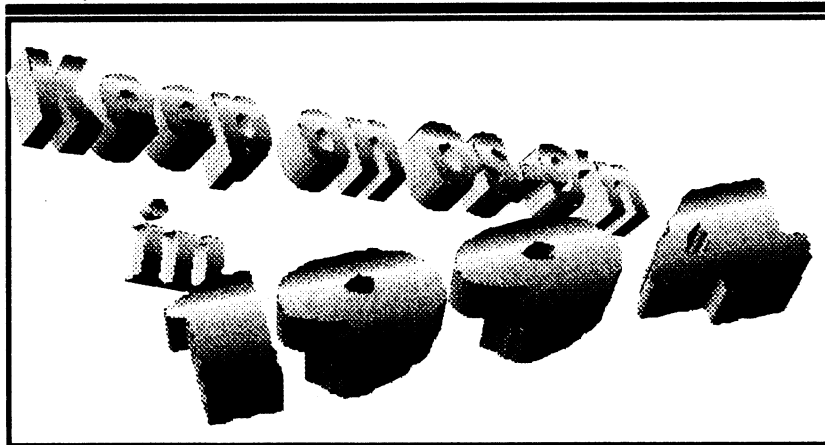
MOST ISSUES ARE \$3.00 (EXCEPT ISSUE #1 WHICH IS \$2.00 AND ISSUE #4/5 WHICH IS \$6.00)

(Shipping is included in all above prices)

Send a check or M.O. to:  
Fat Cat Publications  
4650 Cahuenga Blvd., Ste #7  
Toluca Lake, CA 91602



**FAT CAT  
Publications  
Bookshelf**



## Advertisers Index

Vendor	Page
ColorSystems	IFC
Glenside Color Computer Club	5
BlackHawk Enterprises	8
Bob van der Poel Software	10
JWT Enterprises	14
ARK Systems USA	17
AniMajik Productions - Software	19
Peripheral Technologies	19
OS9 Underground Advertising	20
Northern Xposure	21
Sub-Etha Software	23
Farna Systems	24
Computer Design Services	28
Dirt Cheap Computer Stuff Co.	29
09-Online	33
AniMajik Productions - Graphics	34,37
CoNect	36
FAT CAT Publications Bookshelf	IBC
DELMAR Co.	BC

Let these fine vendors know you saw it in **The Underground!**

# THE "INTERNATIONAL" OS9 UNDERGROUND® MAGAZINE

Dedicated to OS-9/OSK Users Everywhere

VOLUME TWO, ISSUE ONE  
CONTENTS



### The Underground Staff

**Editor/Publisher:**  
Alan Sheltra

**Assistant Editors:**  
Jim Vestal  
Steve Secord

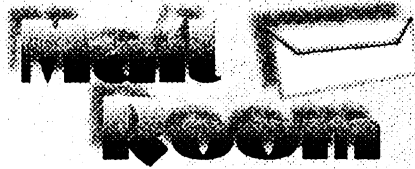
**Technical Editor:**  
Leonard Cassady

**Contributing Editors:**  
Wayne Campbell  
Allen Huffman  
Andy Jacobson  
Scott McGee  
Colin McKay  
Eric Levinson  
Boisy Pitre  
Bob van der Poel

**Typesetting/Layout/Artwork:**  
AniMajik  
Productions

<b>Mall Room</b> (Letters to the Editor)	<b>4</b>
<b>Under It All</b> (Editor's Column)	<b>6</b>
<b>New Discoveries</b> (What's New) by Jim Vestal	<b>7</b>
<b>Real World OS-9</b> A Look at Real-World OS-9 by Andy Jacobson	<b>9</b>
<b>Computer Science 201</b> Lesson Four: Binary Trees by Scott McGee	<b>12</b>
<b>BASIC Training</b> NCode/DCode by Wayne Campbell	<b>18</b>
<b>Creating Your Own C Libraries</b> Part 2 by Bob van der Poel	<b>25</b>
<b>C Software Engineering</b> ANSI C by Leonard Cassady	<b>33</b>
<b>Tic Tac Toe - Score Mod.c</b> by Jim Vestal	<b>36</b>
<b>OS9 Underground Member Card</b>	
<b>Vendor list</b>	<b>37</b>
<b>Shell Game</b> (Cartoon)	<b>38</b>
<b>Advertiser's Index</b>	<b>38</b>

Fat Cat Publications and The "International" OS9 Underground Magazine and its logotypes are registered trademarks. Subscription rates are \$18.00 for 12 issues (\$22.00 Canada, \$27.00 overseas US Funds). Single or back-issues are available at the cover price (please call or write for availability). Fat Cat Publications is located at 4650 Cahuenga Blvd., Ste #7, Toluca Lake, CA 91602. (818) 761-4135 (Voice), (818) 365-0477 (Fax) or (818) 769-1938 (Modem). The contents of these pages are copyrighted. Photocopies or illegal reproduction of this magazine in part or whole is strictly prohibited without prior written permission.



## LETTERS TO THE EDITOR

### Wants to Score a ScoreMod

Dear Editor,

In Issue #10 there was a tic tac toe program. You did not include scoremod.c program listing in that issue and said it would be in the next issue. I haven't seen it yet, will it be in a future issue?

-Keith Bauer, Memominee, MI  
(received from Delphi via AOL internet gateway)

Keith,  
Lack of space prevented the the listing from appearing in issue #11 or #12 as was planned. The listing "ScoreMod.C", however, does appear in this issue (Page 36).

Listings may always be requested via internet (I have forwarded about 6 requests in the last month, Keith was one of them!). See page 15 for information on where to send requests.

-Editor

### Glaring Error

I was just going over Cassidy's article in the Dec. issue and have found a glaring error which should be commented on in the next issue! On page 10 in the discussion on strcat() it states that memory is checked and automatically allocated for the operations. It ain't so...and if one relies on that you'll get wonderful program crashes!

-Bob van der Poel, Port Hill, ID

(Received from CIS via AOL internet gateway)

Comments, Leonard?  
-Editor

### Trashed File

I was just about finished with my article on how I simulate virtual memory and break the 64K barrier under OS9 Level II, but somehow the file was trashed on n coco. So I'm starting from scratch. What a bummer, I was 90% finished. It's really frustrating to get so close to being finished and then lose it. Basicly, I just wanted to let you know I hadn't forgotten. BTW: Your mag is really starting to look professional. Something that might be helpful to the community would be a listing of all ftp and internet listserv sites that support OS9. Currently, I coco@pucc.princeton.edu, and a mailseru at OS9archive@cabrales.cs.wisc.ed Also, I noticed you have an account on America Online. I'm thinking of joining either GENie or America Online. Do you know which has the better OS9 group.

-Chris Strickland,  
New Horizons Software  
(407) 264-9328

Chris,  
Sorry to hear about all that lost wc Chris... yowch! I know how frustrating that can be.

I like the idea of listing ftp sites for OS-9. Look for something in a future issue. Between GENie and AOL, I'd go wi GENie for the better OS-9 Group. I use AOL mainly for email.

-Editor



```
for ( k = 0; k < 3; k++)
if ( a[k][k] == 'X' )
xcount++;
else if ( a[k][k] == 'O' )
ocount++;

return( evaluate(xcount, ocount));
}

int diag2score( a )
char a[[3];
{
int k;
int xcount = 0;
int ocount = 0;

for ( k = 0; k < 3; k++)
if ( a[k][2-k] == 'X' )
xcount++;
else if ( a[k][2-k] == 'O' )
ocount++;

return( evaluate(xcount, ocount));
}
```



So don't just look like the other guys...  
Let us make you **stand out!**

Prices start as low as \$10.00!  
Give us a call and let us know what your needs are...

We make you Look Good!

AniMajik  
Productions®

(818) 761-4135

Animajik has been providing quality graphics and design since 1975....

OS9 Underground  
Magazine  
Member Card

### Participating Vendors

These vendors will offer the following discounts for OS9 Underground Member Card Holders

- CoNect \_\_\_\_\_ 10% Off any order
- Sub-Etha \_\_\_\_\_ 10% Off any order
- Canaware \_\_\_\_\_ 10% Off WristSavers  
10% Off WristSaver MousePad & 15% Off ENC9
- AniMajik Productions \_\_\_\_\_ 20% Off all Software
- Fama Systems \_\_\_\_\_ 10% Off any order
- Bob van der Poel Software \_\_\_\_\_ 10% Off OSK Software

Software/Hardware Vendors you can be listed here FREE!  
Contact the OS9 Underground for details. You need not be an advertiser for this Free service.

Be sure to give your card number when you place an order with these fine vendors. Not responsible for typos or mis-prints

# CoNect New Hardware

**Mini-RS232 Port:** If you are in the market for a Tandy-compatible serial port, check out our Mini! This ROMPak unit supports all seven control lines available, and can supply more output current than even the Tandy Pak! Jumper selectable addresses and cd swap.

**Only \$49.95**

(Y Cable use requires 12 volt power supply, add \$9.95)

**XPander:** The XPander allows you to assemble the most compact CoCo3 system possible using a stock motherboard. For example, the electronics of my 2 meg CoCo3 with Tandy Floppy Controller, Burke & BurkeXT, WD1002 Hard Drive controller, rs232 port, puppo and Hi-Res adaptors froms a block 12 inches long, 7 inches deep and 3 inches at it highest point.. Not only will this fit in the smaller PC cases,, but in a modified CoCo case.

Obviously this is not a full tilt MultiPak clone - there just isn't room. The two external slots may both contain /scs decoded devices, but only one slot ROM may be used. The external slot may be used either as a ROMPak port (disables internal hardware when Pak is inserted), or as an undecoded buss slot. 12v is available at all slots.

The no-slot RS232 port is a virtual clone of the mini-rs232 described above, and saves not only a slot but quite a bit of room in the finished package.

The Xpander is available in two versions. If a PC type case/power supply will be used, order just the board. CoCo Kit includes a new lower case shell and 450ma +/- 12v power supply.

## CoNect Custom CoCo Cables!

Cassette - (mini or rca)	\$ 4.95	Comp. Video (6 feet)	\$ 3.95
Disk Power (either way)	\$ 5.95	Printer (DIN to DIN)	\$ 4.95
2 Drive Floppy Data	\$14.95	3 Drive Floppy Data	\$19.95
RS232 (db 25, db9, din4 to db25,db9 - normal, nulmodem, or dcd swap)			\$ 9.95

## Ram Upgrades

64K CoCo 1 (F board), or CoCo2 (2 or 8 chip) with instructions	\$ 7.95
512K CoCo3 (various makes) not always available	\$49.95
2 Meg CoCo3 (Disto) with SIMMS	\$194.95 Installed (add 11.95 for 6309)

## CoNect

449 South 90th Street  
Milwaukee, WI 53214

(414) 258-2989 • Delphi: RICKULAND • Internet: rickuland@delphi.com

The Glenside Color Computer Club  
of Illinois presents...

# The 2<sup>nd</sup> Annual 'Last' CoCoFest

May 21st & 22nd, 1994

**Holiday Inn Elgin** (A Holidrome Indoor Recreation Center)

345 W. River Road (A city block from I-90 & IL-315)

Elgin, Illinois (Same great location as last year!)

Overnight room rate \$52.00 (plus 10% tax)

Call 1(708) 695-5000 for reservations

Be sure to ask for the "CoCoFest" rate!

Admission is \$15.00 at the door (2-day pass only)

Advance ticket sales \$10.00 + SASE (or \$10. + 50¢ P&H)

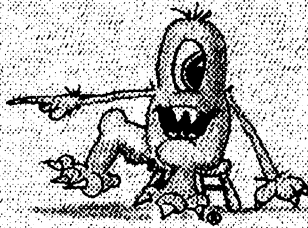
George Schneeweis, Treasurer,  
Glenside Color Computer Club  
RR#2, Box 67, Forrest, IL 61741-9629

Tony Podraza, Fest Chairman  
(708) 428-3576 (Voice)  
(708) 428-0436 BBS  
Delphi: **TonyPodraza**

Carl Boll, Vice President GCCCI  
(312) 735-6087 (Voice)  
(312) 735-3355 (BBS)  
Delphi: **CBJ**

Brian Schubring, President GCCCI  
(708) 529-3539 (voice)  
Delphi: **THESCHU**

# Under Under is All it All



(Editor's Column)

by Alan Sheltra (ZOG)

## New Volume, New Year!

Happy New Year! I sincerely hope 1994 brings you the best.

The Underground starts the new year with our **Anniversary Issue** and with 12 issues under our belt (13 including this one) we're ready to continue to provide the *best* articles, tutorials, news and views of OS-9 for another year.

We intend to continue making improvements and refining this publication even more as time goes on... look for more changes in the near future.

## First Class

I hear one publication is going to 3rd class mail. As a cost conscious editor/publisher, I looked into the idea... and totally rejected it! The Underground will ALWAYS be sent First Class Mail.

The U.S. Postal Service does treat 1st class mail much better than it does 3rd class mail, and the cost difference isn't worth the headaches the cheaper way could cause. Besides, first class mags, gotta go first class ☺!

## TheFatCat List

The Underground has it's own email list (sorta like the CoCoList at Princeton) and everyone is welcome to participate

(it's free, see page 15 for details). V occasionally have contests on-line. In fact we just had 3 winners from our New Years Contest. Congratulations to First Prize winner Timothy A. Johns!

## The Return of the MM/1

David Graham, president **BlackHawk Enterprises**, has announced a special deal to buy an MM/1. He's asking a deposit of only \$10.00 to the first 20 people to order one of his 3 available MM/1 systems. In return for being one of the first 20, he is offering a special discount. Best of luck to David at BlackHawk. (See BlackHawk's ad on page 8.)

The Underground has decided to take advantage of this deal and has send in an advanced order. The MM/1 will make a nice addition (along with o present TC70), to our programmi arsenal and review arsenal.

## Contest

Next month, we'll kick off the year with another contest. Until then, *Keep on doin' it in real-time!*

-ZOG (Alan Sheltra)



(Continued from October 1993 Issue)

## by Jim Vestal

This file contains the function score and other functions that it calls to evaluate the score for a tic-tac-toe square.

To use it, write prog5.c using the prototype extern int score( char[][3], int, int );

Then compile prog5.c and scoremod.c as follows :

cc -Aa -c scoremod.c <- creates scoremod.o

cc -Aa -c prog5.c <- creates prog5.o

cc -o prog5 prog5.o scoremod.o <- links the two .o files and creates the executable prog5

To execute your program, give the command prog5

```
*/
int score( a, r, c )
```

```
char a[][3];
int r;
```

```
int c;
{
```

```
int total = 0;
total += rowscore( a, r );
total += colscore( a, c );
if ( r == c )
total += diag1score( a );
if ( r + c == 2 )
total += diag2score( a );
return( total );
}
```

```
int evaluate(x,oh )
```

```
int x;
int oh;
```

```
{
if ( oh == 2 )
return(50);
if ( x == 2 )
return(25);
```

```
if ( ( oh == 1 ) && ( x == 0 ) )
return(10);
if ( ( x == 1 ) && ( oh == 0 ) )
return(8);
if ( ( x == 0 ) && ( oh == 0 ) )
return(4);
else
return(0);
}
```

```
int rowscore( a, r )
```

```
char a[][3];
int r;
```

```
{
int k;
int xcount = 0;
int ocount = 0;
```

```
for ( k = 0; k < 3; k++ )
if ( a[r][k] == 'X' )
xcount++;
else if ( a[r][k] == 'O' )
ocount++;
```

```
return( evaluate(xcount, ocount));
}
```

```
int colscore( a, c )
```

```
char a[][3];
int c;
```

```
{
int k;
int xcount = 0;
int ocount = 0;
```

```
for ( k = 0; k < 3; k++ )
if ( a[k][c] == 'X' )
xcount++;
else if ( a[k][c] == 'O' )
ocount++;
```

```
return( evaluate(xcount, ocount));
}
```

```
int diag1score(a)
```

```
char a[][3];
```

```
{
int k;
int xcount = 0;
int ocount = 0;
```

Continued Page 37

**EXPRESSIONS**

**Structure assignment**

ANSI C allows the initialization of union variables. If the first component of a union is a structure, the entire structure is initialized.

The initializer for a static, external, automatic, or register union variable must be a brace enclosed constant expression.

**Passing structures as arguments**

There are methods of structure passing, by value and by reference. The ANSI C Standard requires the compiler to support passing structures by value.

It is worth noting that there are only two circumstances where passing a structure by value is useful since the entire structure must be copied.

- The structure is very small.
- To guarantee the called function does not change the structure being passed.

**STORAGE CLASS**

**Function prototypes**

The two main benefits of function prototyping are:

- compiler type-checking of actual function arguments at compile-time.
- automatic argument conversion is turned off.

Prototyping usually reduces the number of conversions needed when passing arguments to functions by allowing specified data types to be passed.

**INITIALIZATION**

**'struct' and 'union' name spaces**

The ANSI C Standard requires the compiler to create a separate 'naming space' within each structure and union, so that two or more structures, or unions may have components with the same name.

This feature is a late extension to the K&R Standard and may not be available on older compilers. Some older compilers do allow 'same naming' of components, but they must be in the same location and of the same data type between the structures or unions.

This is the third installment of several in a series on the additions and differences introduced by the ANSI C Standard.

Any comments, suggestions are welcomed and should be sent to:

Software Engineering Assoc.  
Standard ANSI C Library  
6530 Independence Ave Ste #16  
Canoga Park, CA 91303

or emailed:  
"zog!leonard@abode.ttank.com" (or see page 15 for other mail and email addresses)

-Leonard Cassady



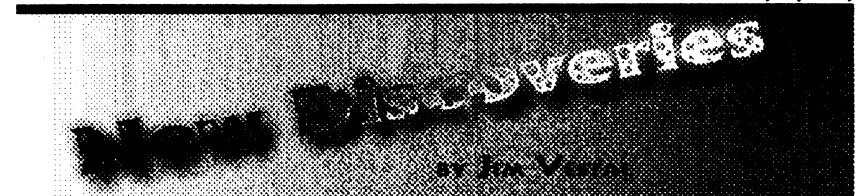
**A NEW SERVICE TO SOFTWARE DEVELOPERS AND DISTRIBUTORS...**

We do something software developers and distributor don't do well... Layout and Design for Manuals, Packaging and Advertising.

*Do you STILL use a dot matrix printer for your manuals and ad copy?*

*Why not let a professional handle your graphics needs... and at prices that won't cost you an arm and a leg...*

Turn to page 37



**NEW DISCOVERIES** CONTAINS A BRIEF OVERVIEW OF NEW PRODUCTS, HARDWARE, SOFTWARE (BOTH COMMERCIAL AND SHAREWARE/PUBLIC DOMAIN). IF YOU ARE A VENDOR OR A SOFTWARE AUTHOR AND WOULD LIKE TO SUBMIT A LIST OF NEW PRODUCTS PLEASE SEND A GENERAL DESCRIPTION TO JIM VESTAL C/O THIS MAGAZINE OR VIA INTERNET EMAIL: [OS9UNDER@AOL.COM, DISCOVER@NARNIA.CITRUS.SAC.CA.US OR ZOGSTER@DELPHI.COM]

**COMMERCIAL PRODUCTS:**

**DDF FROM ARK SYSTEMS USA**

A UNIX-LIKE DEVICE DESCRIPTOR FOR OS9/68000 V2.4. (LOOK FOR A REVIEW OF THIS NEW PRODUCT IN OUR NEXT ISSUE). FOR MORE INFORMATION, PLEASE CONTACT ARK SYSTEMS USA (408) 244-5358. EMAIL: ARKUSA@AOL.COM

**PIPELINE ONLINE**

MICROWARE SYSTEMS HAS RELEASED A TEXT-ONLY VERSION OF THEIR QUARTERLY MAGAZINE PIPELINES TO THE OS-9 COMMUNITY. BEGINNING WITH VOL. 8 NO. 1, PIPELINES IN ARCHIVED FORMAT CAN BE FOUND ON COMPUSERVE AND THE STG NETWORK. SOON IT WILL BE AVAILABLE ON DELPHI AND OTHER MAJOR BULLETIN BOARDS. THUMBS UP TO MICROWARE!

**OS-9 VERSION 3.0 FROM MICROWARE**

MICROWARE ANNOUNCED IT'S RELEASE OF OS-9 VERSION 3.0, WHICH IS AVAILABLE IN TWO KERNELS, STANDARD OS-9 AND ATOMIC OS-9. THE STANDARD OS-9 KERNEL CAN BE USED FOR BOTH RESIDENT DEVELOPEMENT SYSTEMS AND TARGET RUN-TIME SYSTEMS. ATOMIC OS-9, A 100% COMPATIBLE SUBSET OF STANDARD OS-9 IS STRICTLY FOR TARGET RUN-TIME EMBEDDED SYSTEMS. BOTH OFFER HIGHER PERFORMANCE AND LESS SYSTEM OVERHEAD THAN PREVIOUS VERSIONS OF OS-9. DETAILS OF OS-9 VERSION 3.0 IS FEATURED IN VOL. 8 NO. 1 OF PIPELINES. CONTACT MICROWARE AT 1-800-475-9000 FOR YOUR FREE COPY OF PIPELINES.

**THE RETURN OF THE MM/1 - BLACKHAWK ENTERPRISES**

DAVID GRAHAM OF BLACKHAWK ENTERPRISES HAS JUST ANNOUNCED A SPECIAL DEAL FOR THE FIRST 20 PEOPLE TO PUT DOWN A DEPOSIT FOR A NEW MM/1 SYSTEM. A \$10.00 DEPOSIT WILL BE USED TO SECURE A LOAN TO RELEASE 50 READY AND WAITING MM/1'S FROM THE FACTORY. THREE SYSTEMS ARE AVAILABLE, THE DEVELOPERS SYSTEM, THE PROFESSIONAL SYSTEM AND THE EXTENDED KIT. CALL OR WRITE FOR MORE INFORMATION: (405) 234-2347 · P.O. Box 10552, ENID, OK 73706-0552. EMAIL: NIMITZ@DELPHI.COM

**ONLINE SERVICES:**

**DELPHI AND TYMNET**

DELPHI ANNOUNCED THAT YOU CAN NOW START CALLING DELPHI VIA TYMNET AT 6:00 PM INSTEAD OF 7:00 PM WITH NO SURCHARGE. THIS IS GOOD NEWS FOR PEOPLE ON THE WEST COAST WHO WANT TO CONFERENCE WITH THOSE ON THE EAST COAST. TYMNET EVENING HOURS ARE NOW FROM 6:00 PM-6:00 AM IN YOUR LOCAL TIME ZONE.

**SHAREWARE AND PUBLIC DOMAIN:**

**UUTOOLS** UPLOADED BY BOISY PITRE (DELPHI NAME: BOISY) UUE AND UUD ARE A SET OF ENHANCED UUENCODE/UUDECODE TOOLS WHICH CAN DYNAMICALLY SIZE OUTPUT FILES INTO MULTIPLE CHUNKS AND AUTOMATICALLY REASSEMBLE AND DECODE THEM INTO OBJECT FILES. EXECUTABLES FOR BOTH OSK AND OS-9/6809, AND SOURCE CODE/DOCUMENTATION.

CONTINUES

## SHAREWARE/PD SOFTWARE (CONT)

**VEFSHOW** FOR THE MM/1 UPLOADED BY **JOELHEGBERG**:  
VEF PICTURE FILE VIEWER FOR K\*WINDOWS. FINALLY RELEASED OVER A YEAR AFTER JOEL WROTE IT.

**JPEG FAQ** AND SOURCE UPLOADED BY **WRHAMBLER**:  
FREQUENTLY ASKED QUESTIONS AND SOURCES TO CONVERT JPEG IMAGE FILES TO AND FROM A VARIETY OF OTHER IMAGE FILE FORMATS.

**RCIS BBS V.2.3** UPLOADED BY **SROTTINGER**:  
DEMO VERSION OF RCIS BBS FOR OS-9/68000 IS NOW AVAILABLE ON DELPHI.  
SUPPORTS EXTERNAL TRANSFER PROTOCOLS, FULL FEATURED REAL-TIME CONFERENCING,  
NETWORKING, ETC.

## FESTS/SEMINARS

THE THIRD ANNUAL "LAST" CHICAGO CoCo FEST  
MAY 21ST AND 22ND, 1994 AT THE HOLIDAY INN, ELGIN, IL. (SAME LOCATION AS LAST YEAR). OVERNIGHT  
ROOM RATE IS \$52.00 (PLUS TAX). FOR RESERVATIONS CALL (708) 695-5000.  
ADVANCE TICKET SALES ARE \$10.00 + SASE. CALL (708) 529-3539 FOR MOR INFORMATION. BE SURE  
TO MENTION COCOFEST WHEN YOU CALL FOR RESERVATIONS!

PACIFIC NORTHWEST CoCoFEST IV  
THE PNW FEST IS SCHEDULED FOR JUNE 24TH AND 25TH, 1994 IN PORT ORCHARD, WA. CONTACT  
DONALD ZIMMERMAN @ (206) 871-6335 OR WRITE TO 3046 BANNER ROAD S.E., PORT ORCHARD,  
WA 98366 FOR MORE INFORMATION.

IF YOU'D LIKE TO SEE YOUR OS-9/OSK PRODUCT LISTED HERE, (COMMERCIAL OR SHAREWARE),  
CALL (818) 761-4135 OR SEND EMAIL TO: OS9UNDER@AOL.COM OR U.S. MAIL  
TO 4650 CAHUENGA BLVD., STE #7, TOLUCA LAKE, CA 91602

**BlackHawk Enterprises** can now make the MM/1 available to the public. We have 50 systems ready at the factory, and need cash to pay for them, SO For the next three months only, choose from these great values!

**The Developers System**

MM/1 Extended with 1 Meg memory, floppy drive, and Case and power supply. Includes OSK 2.4 with C and Basic, and Microware documentation with all available KWindows documentation.  
Suggested Retail Price: \$1125      *Introductory Price:* **\$975**      **You save: \$150**

**The Professional System**

A Developers system, minus only the Microware documentation!  
Suggested Retail Price: \$1025      *Introductory Price:* **\$875**      **You save: \$150**

**The Extended Kit**

MM/1 Extended board set - no floppy drive, case or power supply. Includes OSK 2.4 with C and Basic, and all available KWindows documentation.  
Suggested Retail Price: \$900      *Introductory Price:* **\$825**      **You save: \$75**

**BlackHawk Enterprises** P.O. Box 10552 Enid, OK 73706-0552

Call (405) 234-2347 from 9 am to 2 pm Central Time

should flagged as a compile-time error.

The second use of the 'void' data type, while commonly seen in coding, is more subtle.

Prior to the ANSI C Standard, it was the programmers responsibility to cast the returned pointer to a pointer to the correct data type. ANSI C implements the 'void' pointer as a 'generic pointer', and is automatically cast to the correct returned pointer type when assigned a value.

Generic pointers add flexibility when using another ANSI C feature called prototyping. When a function has a formal parameter, the actual argument must be of the same type cast. By using the 'void' pointer type, the formal may be of any type cast.

EG:

```
char *strcpy(char *s1, const char *s2); void
*strcpy(void *s1, const void *s2);
```

The only consideration when using the 'void' data type in this manner is function variable storage allocation. In the above example, the function 'strcpy()' expects a 'char' data type and variable storage limits may be overrun if the value is larger than a 'char' data type.

**'enumeration' data types**

The enumerated data type is a recent addition to the K&R Standard and other Tradition C implementations. Similar concepts to this data type occur in other languages, such as Pascal and Ada.

Enumerated types are implemented by associating integer values with the enumerated constants. The first enumerated constant receives the value of zero, (0), if no explicit value is defined. Subsequent enumerated constants without explicit value association, receive an integer value one greater than the previous constant.

The ANSI C Standard specifies that enumerated types must be of an 'int' type cast. This strict type checking is not always present in pre-ANSI C compilers

and could introduce run-time errors into the program.

**STATEMENTS****'switch' expression control**

The K&R Standard requires the switch statement expression to be of an 'int' data type.

The ANSI C Standard requires the expression only to be an integral expression and allows data types of 'char', 'short', 'int', or 'long' and the number of case-label expressions supported must be a maximum of 257.

**09-Online Systems**

Technology Concepts  
LineLink 14.4 v.42bis  
Data/Fax External Modem

**Only \$125.00!**  
(Price Includes shipping)

Send check/M.O. made payable to: J. E. Vestal

**09-Online Systems**  
c/o Jim Vestal  
221 E. 17th. #31  
Marysville, CA 95901  
(916) 743-4264

Ask to be placed on our mailing list for a Free shareware catalog.



Example:

```
int const *pointer;
```

This defines a pointer to a 'const' data object and ensures the object cannot be modified. However, the pointer address may be reassigned to a different 'const' data object address. Conversely, passing a 'const' pointer to object argument:

```
int *const pointer;
```

ensures the object 'pointed to' cannot be modified and must point to the same 'const' object for its entire duration.

### 'volatile' qualifier

The 'volatile' qualifier informs the compiler that the data object may be modified in ways the compiler may not recognize and turns off optimization. This ensures the compiler translation of the source code is exactly as it appears in the file and not replaced by the compilers' idea of a more efficient algorithm.

### INTEGER DATA TYPES

ANSI C compiler implementations are required to document the ranges of integers in the include header file, 'limits.h'. The compiler is free to substitute its own values, but must support the minimum values and must have the same sign extension as specified in the standard.

These restrictions support systems using the three most common types of binary integer encoding techniques, (internal integer representation).

- Two's Complement Notation.
- One's Complement Notation.
- Sign Magnitude Notation.

### 'signed' data types

ANSI C implemented the 'signed' data type as part of the standard, having a value range of (-32,767 to +32,767). This allows variables to hold either positive or negative values. Most compilers assign

variables as 'signed' data types by default, so use of the 'signed' keyword is usually for informational purposes since the three common notations represent integers identically.

The one exception is the 'char' data type. Your compiler will most likely use a 'signed' char data type by default, however, this determination is left to the compiler developer and might produce unusual results or side effects when using the type 'char', or bit fields in structure. Consult your compiler documentation for the default data type.

### 'unsigned' data types

Each 'signed' integer data type has a corresponding 'unsigned' data type. The 'unsigned' data type is restricted to non-negative values and have double the positive range of the corresponding 'signed' data type, (0 to 65,535). The storage requirements are the same for 'signed' and 'unsigned' integers, however the integer encoding techniques between the two data types may be different. Straight binary notation is used to represent 'unsigned' data types, regardless of the 'signed' notation technique.

### 'void' data type

The 'void' data type was not originally part of the C language. The semantics were vague and varied across the many implementations. The ANSI C Standard defines two qualities that must be supported, although the compiler developers are free to include others.

The first use of the 'void' data type, and the most commonly used, is to indicate that a function does not return value, or returns a NULL pointer.

```
EG: extern void foobar();
void foobar( a, b ) int a; char b; { ; }
```

This informs the compiler that any attempt to use the returned value the function 'foobar()' is a mistake and

# Real World OS-9

First, let me introduce myself. My name is Colin McKay, and I am the Executive Vice President of the OS-9 Users Group, Inc.

I write a regular column for the Users Group newsletter, the MOTD. The column, "Straight from the Horse's Mouth", is about the use of OS-9 in Educational, Industrial and Scientific institutions. The following article is a reprint from that column.

Opinions expressed herein are those of the author, and may or may not reflect those of the OS-9 User Group, or the OS9 Underground.

*Dept. Pharmacology / Div. Nuclear Medicine and Biophysics UCLA School of Medicine*

Andy Jacobson is a graduate student in Pharmacology at the University of California in Los Angeles (UCLA). There, he has become a jack-of-all-trades, and has become involved in cell biology, organic chemistry, physical chemistry, too much microscopy and scientific visualization, and some C programming. Pretty far from classical pharmacology, though he does use a lot of pharmacological techniques in his work.

Although he does not come from a computer science background, Andy is involved with setting up the system we will be talking about, administers a Sun, and work with several other computer systems.

One of the systems that they use is a Leica Lasertechnik Confocal Laser Scanning Microscope (CLSM), which uses

by Andy Jacobson

Intro by Colin McKay

Permission to print this article was granted by the OS-9 Users Group.

the run-time version of Professional OS-9 v2.4.

Confocal is a light microscopy technique which differs from standard optical microscopes that we are all familiar with. This method allows somewhat higher resolution and reduced depth of field by focusing light onto (and from) a single point in the sample at a time, allowing for 3-space localization of signal. The signal is recorded in a raster scan, which is where OS-9 comes into the picture, to control the scan mirrors, and collect the data.

The CLSM is used rather than an Electron Microscope for a couple of reasons. With a confocal microscope you can use:

- a) whole cells, not just slices;
- b) live cells, to do dynamic (time) studies;
- c) multiple labels (input and fluorescent light wavelengths can be filtered).

A project currently underway using the CLSM involves looking at models of

how chemical receptors produced by cells are regulated, to better understand how they are affected in Parkinson's disease. There are many people using similar techniques to study new drug development and environmental effects on cells.

The CLSM is used examine intracellular localization of dopamine receptors transfected into a mouse fibroblast cell line. Now for the English translation:

Dopamine is an amino acid, and is one of many chemicals (neurotransmitters) which transmit signals in the brain. Improper use of dopamine by cells certain portions of the brain is related to Parkinson's Disease. The recent movie 'Awakenings' dealt with a similar subject.

Genes that produce dopamine receptors are spliced into the the chromosomes of mouse fibroblast cells. The gene is from rat brain, while the host cells are mouse fibroblasts (sort of undifferentiated connective tissue cells), which have no receptors of their own.

## Bob van der Poel Software

### Great Stuff for OSK

**Ved Text Editor** - The most powerful editor for OSK. Latest version includes an integrated spelling checker and text formatter. Complete editing functions. Includes MVEF for changing Ved environment files. Works on any terminal. \$59.95.

**Vprint Text Formatter** - The ideal companion for Ved, this command based formatter supports proportional fonts, a powerful macro language, complete margin/indentation control and footnotes. \$59.95.

**Vmail** - Written to manage our own mailing lists, this mailing list program is slick and fast. Using a proprietary system of linked lists and external indexes any name can be found and printed instantly. \$49.95.

**Cribbage** - The classic card game! Pit your skill against the computer. Requires Kwindows and mouse. \$19.95.

**Magazine Index System** - Find those magazine articles...fast. File compatible with our Level II product. Comes complete with 300K of Coco files. \$19.95.

All prices shown are for single user licences. Please write for multi-user pricing or a free catalogue (Level II or OSK). Please add \$3.00 shipping and handling to all orders (\$5.00 overseas). To order, please send your check or money order and preferred disk format to:

**Bob van der Poel Software**  
 P.O. Box 57                      P.O. Box 355  
 Wynndel, BC                    Porthill, ID  
 Canada V0B 2N0                USA 83853  
 Telephone 604-866-5772

## Byte length

The ANSI C Standard requires a byte to be at least eight bits in length and integers to be at least sixteen bits, (four bytes), long. The Standard also requires integers to be the 'natural size', or the number of bits the CPU uses to handle a single instruction.

This requirement places the ANSI C Standard in conflict with the older 8-bit CPU processors as the 'natural size' of an integer is usually two bytes in length for these systems. When this conflict arises, the 'natural size' takes precedent over the 8-bit byte requirement.

## Integral data type ranges

The compiler dictates certain constraints on the programs it will compile. Translation and numerical limitations represent how the compiler will interrupt typed data objects and constructs. ANSI C compilers must support specified minimum values, but are free to exceed the maximum limitations.

Numerical limits are the range of values of each scalar object type an ANSI C compiler must support and these are defined in the 'limits.h' and 'float.h' include header files.

ANSI C compilers must support the following translation limits:

- 15 nested levels of iteration and control structures, (compound statements).
- 6 nested levels in conditional compilation.
- 12 pointer, array, and function basic-type modifiers in the declaration.
- 127 parentheses-nested expressions.
- 31 significant characters in macro names and internal identifiers.
- 6 significant character in an external identifier.
- 511 external identifiers in one source file.
- 127 identifiers within the scope of one program or function block.
- 1024 macro names within one source file.

- 31 parameters in one function call or definition.
- 31 parameters in one macro definition.
- 509 characters in a source line.
- 509 characters in a string literal (after concatenation).
- 32767 bytes in an array or structure array.
- 8 nested levels for '#include' files.
- 257 'case' labels in a 'switch' statement.

## TYPE SPECIFIERS AND QUALIFIERS

Type specifiers provide information or definitions about the data types of the identifier names. The side effects may occur when data types are defined as type tags, structure and union component names, and enumeration constants.

Declaration syntax requires a storage class specifier, a type specifier, a type qualifier, or some combination of these three. Older compilers often did not implement type specifiers, or omitted the 'void' type qualifier and maintenance and readability of source code suffered.

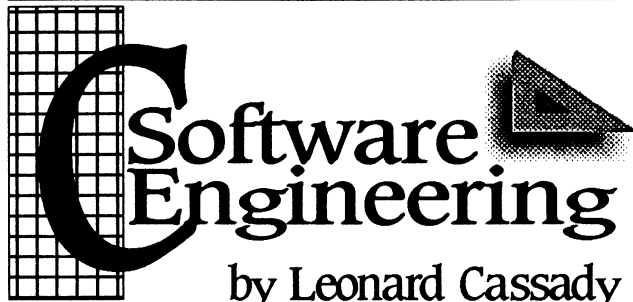
Generally, type specifiers indicate the storage class of return types in functions and denote the various possible side effects of operations on variables. The default type specifier for variables and functions is an 'int' type cast unless otherwise 'specified'.

ANSI C introduced the type qualifiers, 'const' and 'volatile' as part of the standard language. They designate data objects whose values are unchanging and whose value may be altered in ways unknown to the compiler.

### 'const' qualifier

The 'const' type qualifier specifies a variable that may NOT be modified in any way once it has been initialized and may be used in place of the preprocessor '#define' directive to identify read-only data.

The 'const' qualifier can produce subtle side effects, depending on its appearance in the definition.



# Software Engineering

by Leonard Cassady

## ANSI C

**L**ast installment, we discussed source translation differences and additions, (lexical analysis), between Traditional C and the ANSI C implementations. This issue, we continue the discussion with data type differences. Next issue we will begin the actual construction of the ANSI Standard Library and the contents of the include header files.

### IDENTIFIER NAMES

One of the most important features of any high-level programming language is the capacity to classify data type. There are eight identifier names that may be declared in C Language.

- preprocessor macros
- statement labels
- type qualifiers
- type tags
- variables
- functions
- enumeration constants
- structure and union components

Preprocessor macros are declared by the '#defin' preprocessor directive, and statement labels are declared by their appearance in a C function or source file. Type tag structure and union components, and enumeration constants are 'type specifiers' in declarations. Variables, functions and types are 'declarators' that appear in declarations.

It then becomes possible to study what the dopamine receptor does on its own, without having to worry about all the other receptor systems in the typical neuronal cell.

The receptors are marked with fluorescent dyes that tend to stain the receptors, and leave the rest of the cell structure relatively unmarked. Also used are other fluorochromes that label cell membranes and other organelles (specialized areas within a cell, roughly analogous to organs), which provide a reference for the location of the receptors.

To make changes to the run-time OS-9 software that is provided by Leica, several packages have been added. From Microware, the Pharmacology Department has added the development pak, NFS (which they found inadequate for their needs), and SrcDbg, as well as mshell. A lot of the software from the TOP package, as well as some of the EFO tools are also used. GCC may also be added in the future.

Changes made were primarily to give control of the login banner, add an encrypted password file, some utmp functions, etc. The Bourne shell, 'sh', is used for almost everything, along with a number of the utilities from Cabre. Microware's umacs is used as the editor.

A modified ftpdc from the good folks at CERN that uses encrypted passwords is also used.

Some of the areas that they still hope to change include:

- expanded user/group id size (for Unix compatibility)
- something that really works like tset that will talk to telnet clients for terminal setup.
- a complete csh
- a new Network File Server (NFS) that better meets their needs
- something like colored memory for swap
- Mac floppy support.

(If you can help out in any of these areas, contact me, and I'll pass on your suggestion or address.)

Hardware used is a MC6803 (MVME 147, w/MVME 147SA-2). There's also an IPP-1 and APAL board from Eltec, some more memory, a Fujitsu 1G dis with a custom driver. Also used are some VME cards from Leica, which run the scanning, Analog to Digital Converter (ADC) and video.

AVS is used for data visualization, on a Sparc10, with some customized software. Animation work is done on the rendering, and transferred to video tape via an Abekas (pronounced abacus). This is essentially a fast hard disk, used in video production to dump frames to video tape in real time.

In addition to the confocal microscope, PET (Positron Emission Tomography) is also used, although that is another story. The department is also involved in the design and synthesis of a new series of environmentally-sensitive fluorophores.

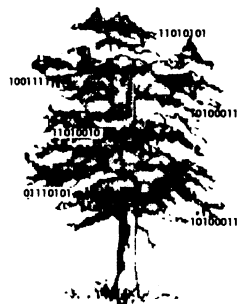
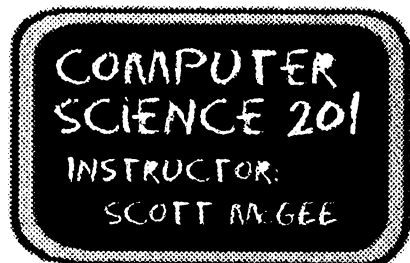
Upcoming columns in future issues of the MOTD include articles about the use of OS-9 by NASA, the Canadian Space Agency, and MIT. MOTD is published 4-6 times per year.

- Colin McKay



If you are interested in joining the OS-9 Users Group and receiving the MOTD, membership details are as follows:

United States and Canada \$25.00 US  
 All other countries \$30.00 US  
 The OS-9 Users Group, Inc. 615  
 W. 63rd St. Suite 109 Chicago, IL  
 USA 60638

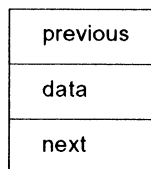


## LESSON 4: BINARY TREES

Last time, while discussing doubly linked lists, we used the following definition for a node:

```
typedef struct DLLIST *dllist;
struct DLLIST{
  int data;
  dllist next;
  dllist previous;
}
```

Graphically, each node would look like this:



As I said last time, we are going to modify this ever so slightly and use it to make an entirely new data structure. In the definition, there are two pointers that point to other nodes (next and previous). We will keep them both, but just rename them. In the following definition, the only other change is to the name we give the types.

```
typedef struct T_NODE *t_node;
struct T_NODE{
  int data;
  t_node left;
  t_node right;
}
```

Scott McGee can be reached through this magazine at the addresses (Email and regular mail) listed on page 15, or email him directly at [smgee@microware.com](mailto:smgee@microware.com)

## Dynamic Systems Presents

### Speedisk

Version 1.0

Single user version only \$99.95

Speedisk is a complete disk optimizer/defragmentor for OS-9/68000

- Runs under ALL versions of OS-9/68000, including Ver 3.0
- Requires no modifications to system software (RBF)
- Understands all aspects of the file system including variable sector sizes, any cluster size, and linked lists.
- Full screen display to monitor progress
- Intelligent file/directory placement to minimize file search time and reduce future fragmentation problems
- Over 3 years in development

All orders add \$5.00 for shipping. Checks or money orders only please  
Call for details

**Dirt Cheap Computer Stuff Co.**

*"Cheap but not trash"*  
1368 Old Hwy 50 East  
Union, Missouri 63084  
314-583-1168

```
* return true if char is 32..126 and NOT(a..z, A..Z or 0..9)
f_ispunct: bsr.b f_isprint
          tst.l d0
          beq.b f_ispunct1 not printable, can't
          *           be punctuation!

          bsr.b f_isalnum
          tst.l d0
          bne fall    if it is alphanumeric, it's not punctuation
          moveq #1,d0
f_ispunct1 rts
-----
* return true if arg is SPACE, TAB, LF, CR, or FF

f_iswhite: cmpl.l #32,d0 space?
          beq.b f_iswhite1
          cmpl.l #7,d0 tab?
          beq.b f_iswhite1
          cmpl.l #10,d0 lf?
          beq.b f_iswhite1
          cmpl.l #13,d0 cr?
          beq.b f_iswhite1
          cmpl.l #12,d0 ff?
          bne.b fall
f_iswhite1 rts
-----
#endasm
```

You might have a look at a neat little shortcut in char\_conv.c. Instead of creating my own psect for an assembly language file, I have placed a #asm/#endasm around the assembler code. This way, the compiler takes care of creating the psect instructions.

All the routines in char\_conv.c are called in the same way as the equivalent standard functions. It is just a matter of inserting a 'f\_' in front of them.

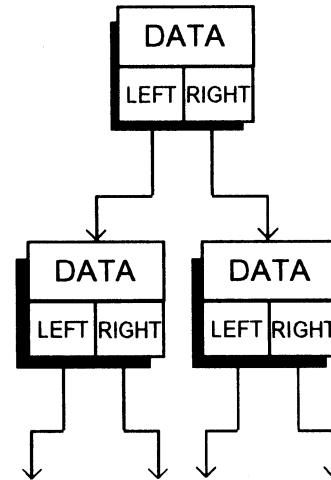
Next month we'll continue on the same theme by presenting some more routines from our library. Don't forget to write...we can be reached at:

PO Box 355,  
Porthill, ID 83853 or  
PO Box 57,  
Wynndel, BC, Canada V0B 2N0 or  
CompuServe :76510,2203.

-Bob van der Poel



Now, you are no doubt wondering why bother changing next and previous to left and right. Well, the reason is, we are not going to have one pointing back to the node that points to us, but will point to **TWO** new nodes, instead of one. This way, instead of creating a one dimensional list, we get a two dimensional tree. Graphically, it would look like this:



To discuss trees, we need to define some terms. One that I have already used is **NODE**. A node is one element in the tree. A node may either be **NULL** (an empty node) or will be a **T\_NODE** structure as defined above. Next is **ROOT**.

The root of a tree is the "first" node in the tree, much like the first node in a linked list is the head. The next two terms, **PARENT** and **CHILD** are used to define the relationship between nodes. In the diagram above, the top (or root) node is the parent node of both of the lower (or child) nodes. The root has no parent. The two lower nodes are both child nodes of the root.

We can differentiate between them by calling them the left child and the right child. (Now you know why we call the pointers left and right.) If two nodes share

the same parent, then they are called **SIBLING** nodes.

Another useful definition is **LEVEL**, this refers to the depth of the tree. For instance, in the tree shown above, there are two levels displayed. We can also use level to indicate the depth of a particular node. The convention I will use is to call the root node level 1. Its children will be level 2, and so on. A **LEAF** or **TERMINAL NODE** is a node in the tree that has NO children.

An **INTERIOR** node is one that **DOES** have children (since it occurs in the interior of the tree. A node and all the nodes below it is called either a **BRANCH** or a **SUBTREE**.

Ok, now that we agree on some terminology, lets continue with our discussion. Lets talk a little about using trees before we consider any code. Like a linked list, a tree may be random, or it may be ordered. If the tree is ordered, due to its more complex structure, there are a **LOT** more ways to order it. A list is sorted in either ascending or descending order. A tree, however, is not linear, and so does not have to have a linear order like that. There is one type of tree, the **Binary Sort Tree**, that does have such an order. Other types of tree structures, however, may take on other types of ordering schemes. In fact, while the tree structure we defined has two children for each node (hence, it is called a **binary tree**), there is no reason that a tree couldn't have more. In fact, one common implementation is to have a structure like this:

```
typedef struct T2_NODE *t2_node;
struct T2_NODE{
    int data;
    t2_node sibling;
    t2_node child;
}
```

Continued Page 15

### NEW 68020 COMPUTER BOARD!!!

The CD68X20 *sizzles* at 25MHZ - processing the most complex calculations in a flash!!!

CD68X20-25, OK RAM	\$699.00
Professional OS9/020 V2.4	\$499.00
ULTRA C Compiler for OS9	\$299.00
SCULPTOR V1.14:6 for Business Software Development	\$ 79.00

Systems Available!

Computer Design Services  
2550 Sandy Plains Rd. Ste. 320-234  
Marietta, GA 30066  
404/973-2170

# JWT Enterprises

## Optimize Utility Set 1:

→ Optimize your floppies and hard drives quickly and easily! → Includes utility to check file and directory fragmentation.  
 → Works alone or with Burke & Burke *repack* utility. → One stop optimization for any level 2 OS-9 system.  
**\$29.95; Foreign Postage, add \$3.00**

## Optimize Utility Set 2:

→ Check and correct any disk's file and directory structures without any technical mumbo-jumbo.  
 → Run periodically to maintain the integrity of your disks as well as the reliability of your data  
 → Especially useful before optimizing your disks.  
**\$19.95; Foreign Postage, add \$3.00**

## Optimize Utility Set Pac:

→ Get both packages together and save!  
**\$39.95; Foreign Postage, add \$4.00**

## Nine-Times:

In each issue:

- The bi-monthly disk magazine for OS-9 Level 2
  - Helpful and useful programs
  - C and Basic09 programming examples.
  - Hints, Help Columns, and informative articles
  - All graphic/joystick interface
  - Can be used with a hard disk or ram disk
- One Year Subscription, \$34.95;  
 Canadian Orders, add \$1.00; Foreign Orders, add \$8.00*

## Back-Issues:

From May 1989, write for back issue contents  
**\$7.00 each; Foreign Orders, add \$2.00 each**

## Magazine Source:

Full Basic09 code and documentation for the presentation shell used with Nine-Times.  
**\$25.95, Foreign Orders, add \$5.00**

NINE TIMES

Technical Assistance & Inquiries  
 (216) 758-7694

JWT Enterprises  
 5755 Lockwood Blvd.  
 Youngstown, OH 44512



Foreign postage excludes U.S. Territories and Canada. These products for OS 9 Level 2 on the CoCo 3. Sorry, no C.O.D.'s or credit cards; Foreign & Canadian orders, please use U.S. money orders. U.S. checks, allow 4 weeks for receipt of order. Ohio residents, please add 6% sales tax.

Copyright (c) 1992

OS 9 is a trademark of Microsoft Systems Corp. and Motorola, Inc.

```
#asm
.....
* Convert character to upper/lower case
f_toupper: move.l d0,-(sp)
        bsr.b f_islower
        tst.b d0
        beq.b f_tolower1 not a lowercase to
start with, do nothing
        andi.b #$df,3(sp)
        bra.b f_tolower1
f_tolower: move.l d0,-(sp)
        bsr.b f_isupper
        tst.b d0
        beq.b f_tolower1
        ori.b #$20,3(sp)
f_tolower1 move.l (sp)+,d0
        rts
.....
* These are fast versions of isupper() and
* islower(), etc.. The standard library uses a
* table to do this inline. However, these
* routines are faster, and shorter. Also, they
* work for unsigned char, which the std.
* library does not:
.....
* return true if arg is a..z or A..Z
f_isalpha: bsr.b f_tolower convert to
lowercase
        bra.b f_islower see if lowercase
.....
* return true if arg is 0..9
f_isdigit: cmpi.l #'0',d0 see if in range 0..9
        blo.b fail
        cmpi.l #'9',d0
        bra.b f_islower1 branch to "bhi.b fail"
instruction
.....
* return true if arg is 127 or 0..31
f_isctrl: cmpi.l #127,d0
        beq.b f_isxdigit3 to RTS
        cmpi.l #31,d0
        bra.b f_islower1 branch to 'bhi fail'
instruction
.....
```

```
* return true if arg is 32..126
f_isprint: cmpi.l #32,d0
        blo.b fail
        cmpi.l #126,d0
        bra.b f_islower1 branch to "bhi.b fail"
instruction
.....
* return true if arg is A..Z
f_isupper: cmpi.l #'A',d0 if in range A..Z
return !0
        blo.b fail too low, return 0
        cmpi.l #'Z',d0
        bra.b f_islower1
f_islower: cmpi.l #'a',d0
        blo.b fail
        cmpi.l #'z',d0
f_islower1 bhi.b fail
        rts return non-zero value (true)
fail moveq #0,d0 return 0 value (false)
        rts
.....
* return true if arg is a..z, A..Z, 0..9
f_isalnum: move.l d0,-(sp)
        bsr.b f_isdigit see if digit 0..9
        tst.b d0
        bne.b f_isxdigit2 is digit, exit (clean stack!!)
        move.l (sp)+,d0 get old d0 back
        bra.b f_isalpha and check to see if it's alpha
.....
* return true if arg is 0..9, a..f, A..F
f_isxdigit: move.l d0,-(sp) save copy of d0
        bsr.b f_isdigit see if it's 0..9
        tst.b d0
        bne.b f_isxdigit2 it is 0..9, exit
        move.l (sp),d0 get original d0
        bsr.w f_toupper convert to
uppercase
        cmpi.l #'A',d0 see if A..Z
        blo.b f_isxdigit1 nope, return 0
        cmpi.l #'F',d0
        bts.b f_isxdigit2 return true
f_isxdigit1 bsr.b fail
f_isxdigit2 addq.l #4,sp restore stack
f_isxdigit3 rts
.....
```

not, drop a note and we'll see about doing a series on using make. In addition to the files, I also have an #include file, "bvlib.h" which contains the needed declarations for my functions. I usually #include this file in all my programs.

```

/* bvlib.h */

/* These are routines from "case_conv.c".
They replace the standard stuff
from ctype.h. These are faster and are
functions, (not macros). Plus,
they work with any integer arguments. */

int f_toupper(), f_tolower(), f_isalnum(),
f_isalpha(), f_isascii();
int f_isctrl(), f_islower(), f_isprint(), f_ispunct(),
f_isupper();
int f_isxdigit(), f_iswhite();

/* Additional string functions */

char *strsave(), *strend();

/* memory functions */

void *memchr();

/* Useful macros */

#define dswitch(a,b)
switch((tolower(a)<<8)+tolower(b))
#define dcase(a,b) case((a<<8)+b)
#define arraysize(a) (sizeof(a)/sizeof((a[0])))
#define offsetof(t, m) (&((t *)0)->m)

```

Any program needing my library functions just needs a library specification in the makefile.

For example, in the makefile for one program I have the following lines, (these are excerpts):

```

LFLAGS = -q -l=/dd/lib/termib.l -l=/dd/lib/bvlib.l -m=16k ...
$(OFILE): $(RFILES) $(CC) $(RFILES) $(LFLAGS)

```

Now, our first module! The "standard" libraries usually contain a set of character conversion and classification routines like: isupper(), islower(), toupper(), etc. They are normally implemented as macros. For a lot of programming they are just fine, however, they can cause problems. First, the functions only work for characters in the range 0 to 128 and the special character EOF (-1). If you are writing programs using unsigned characters you can't use the standard functions without a lot of problems.

Second, the fact that the routines are implemented as macros can create subtle problems.

The module, "char\_conv.c", implements all of the standard routines as functions. To make these functions as fast and short as possible, I have done the coding in assembler. However, you could do the same in portable C and lose very little. Frankly, doing this in assembler was a good excuse to learn 68000 assembler.

```

/* char_conv.c */

/* This file contains fast version of the
standard character conversion
routines and the is??() routines. They are
much faster than the table lookup
routines supplied in a standard library. They
also handle characters in the
complete range of integers..the standard
library routines return incorrect
data for characters outside the range 0..127

```

If you use these routines, don't include <ctype.h> in your sources, use the <bvlib.h> file instead...

```

*/

```

Continued from Page 13

With this type of node structure, sibling is basically a linked list of sibling nodes. If sibling is not NULL, then the node has more siblings and this points to the first of them. We will discuss this type of tree in a future article, but for now, lets return to our binary tree.



Keeping in mind the things we learned from working with linked lists (*special handling for insertion/deletion from the head or tail [root or leaf for a tree], and such*) lets consider insertion into a tree.

First, we need to decide what sort of ordering (*if any*) our tree should have.

If a node has a variable number of children, it may be very useful without any ordering imposed on it. A directory tree on disk is just such a tree structure. If we are using a binary tree, however, we probably should impose an order on it to aid in locating items within it. I mentioned the **Binary Search Tree** earlier, and we will deal with that now

The ordering that makes a **Binary Search Tree** work goes like this. The value in a node determines which subtree any further values will be in. In other words, if a node contains x as its value, and we are looking for a node that contains y, then it is in one subtree if x<y and the other if y<x. For this particular tree, we will specify that the left subtree contains only values less than that in the node, and the right subtree contains only values that are greater than that in the node. We will reserve discussion of duplicate values for later.

Now, to make sure we understand how we want this to work, lets look an example of inserting into a tree before we

## How to reach The OS9 Underground:

For subscription information, questions, call or write to:

**Fat Cat Publications**  
**4650 Cahuenga Blvd, Ste #7**  
**Toluca Lake, CA 91602**  
**(818) 761-4135 (voice)**  
**(818) 365-0477 (fax)**  
**(818) 769-1938 (BBS)**  
 or by email to:  
**EditorOS9U@AOL.com**  
**ZOGster@Delphi.com**

Article Submissions may be sent to the above addresses or to the following special internet account set up for that purpose:

**OS9Under@AOL.com**

Submissions should be sent as plain text and unformatted.  
 Join us on the New OS9 Underground  
 Email, Mailing List

**TheFatCat@AOL.com**

To Subscribe, just send email to the above account and you will be added to the list

Change of Address:

If you anticipate a change of address, you must notify Fat Cat Publications at least 30 days prior your move to continue uninterrupted service. We DO NOT re-send returned mailed. To re-claim returned mail, you must send \$1.00 postage and handling per issue to the above address.

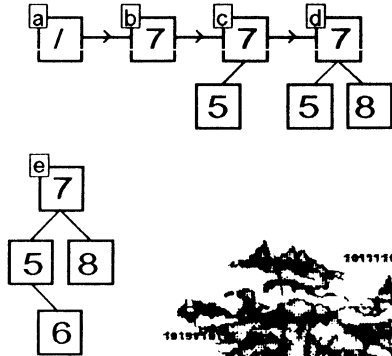
Back Issues:

Back issues are available at the cover price. Call or write for more info.



**Fat Cat  
Publications**

deal with code. We'll start with an empty tree, and insert the following values: 7, 5, 8, and 6. The tree will go through the following changes:



In **a**, the tree is empty. In **b**, we insert 7 so it appears at the root. In **c**, we insert 5. Since 5 is less than 7, it goes in the left child. In **d**, we insert 8. 8 is greater than 7 so it goes in the right child. In **e**, we insert 6, which is less than 7 and so goes in the left subtree. Since the left subtree is not null, we look at its root which is 5. 6 is greater than 5 so we insert it in the right child. If we went on to insert 2, it would appear as the left child of 5. Similarly, if we insert 11, it would appear as the right child of 8.

As you can guess from the above sequence, the actual shape of the tree is dependent on both the data it holds and the order in which it was added. You might try building the tree made by adding 5, 7, 8, and 6 to an empty tree.

Now, let's discuss an algorithm to accomplish these insertions.

```

b-tree insert:
root = tree
while root != NULL
  if value < root->data
    if root->left != NULL
      root = root->left
    else
      root->left = newNode(value)
      root = NULL
  else
    if root->right != NULL
      root = root->right
    else
      root->right = newNode(value)
      root = NULL
    
```

What we are doing here is to assign the variable root to the root of the tree. From now on, it will point to the sub tree in which we know the value belongs. The while loop will step root down the tree, branching either left or right as appropriate, until a null node is found. This node, since it is empty, must be the correct location for the value, so a new node is created with value as its data and is assigned there. Finally, root is assigned the value NULL to terminate the loop.

Notice that if tree is NULL, nothing will happen here. This means that we will have to add code to detect an empty tree and assign the root node.

This will be the homework assignment for this session, along with converting the pseudo code above to functioning code. Next time, we will discuss deletion from the tree and try to figure out what to do with duplicate values.

**Happy New Year!**  
**Class Dismissed.**

-Scott McGee



# Building Your Own C Libraries - Part 2

by Bob van der Poel

**L**ast month we started looking at C libraries. If you've forgotten our discussion you might want to do a quick review since we are going to get right into the programming stuff! As you develop programs you will find that you are using the same routines over and over again. Of course, you could just grab the source code for these routines and compile it with your program...but placing it all in a library is much easier. You could create a library of highly specialized routines, (perhaps for statistics, data bases, graphics, etc), or you can just place a number of your own functions into a personal library. This is what I've done.

I use the "make" utility to maintain my library. Following is a copy of my "makefile".

```

#Makefile for bvlib
TEMP = /r0
CC = cc
CFLAGS = -I$(TEMP) -s -q
OFILE = /dd/lib/bvlib.l
RDIR = .RELS

RFILES = memchr.r bsearch.r writeln.r creadln.r fgets_nl.r lowstr.l
upstr.r rnd.r memcount.r strend.r strtime.r strucmp.r struncmp.r
strpad.r strstr.r strsave.r strstr.r ptr_strcmp.r char_conv.r

$(OFILE): $(RFILES)
    chd $(RDIR); merge $(RFILES) >$(OFILE)
    
```

This makefile contains the names of the various modules in the library and the specific instructions to create the library, (you'll have to change this a bit for Level II). Hopefully, you are familiar enough with the make utility to understand the file...if



# Tandy's Little Wonder

*the most complete reference ever written for the Color Computer!*

**This 140 page softbound book contains:**

- History of the CoCo
- Club and BBS Listings
- Current Supporting Vendors
- Peripheral Details
- Operating System Descriptions
- Programming Languages
- Repair/Upgrade/Modification Procedures
- Schematics (reprinted w/permission of Tandy)
- MUCH, MUCH MORE!

**ONLY \$25 (+ \$2.50 S&H)**

(Canadians add \$2 for air mail, overseas add \$4)

**Introducing a NEW MAGAZINE for CoCo/OS-9/OSK users:**

the world of  
**68' micros**  
Tandy Color Computer, OS-9, OSK

Where does one now go for CoCo support since "the Rainbow" ceased publication? "the world of 68' micros is dedicated to producing a quality publication supporting the CoCo, Disk BASIC, 6809/OS-9, and even OSK (OS-9/68000)! Top writers and articles will be featured, including a hardware column by the infamous **Dr. Marty Goodman**. Upcoming features will include:

- Repackaging the CoCo** (even a transportable!)
- C Programming for Beginners**
- Beginning OS-9... from the box!**
- CoCoFest Reports...** FOUR this year!
- MicroNews...** new products and information (w/ photo of the B&B "Rocket")
- Swap Shop...** classified ads! (Subscribers only, buy,sell trade... even software!)

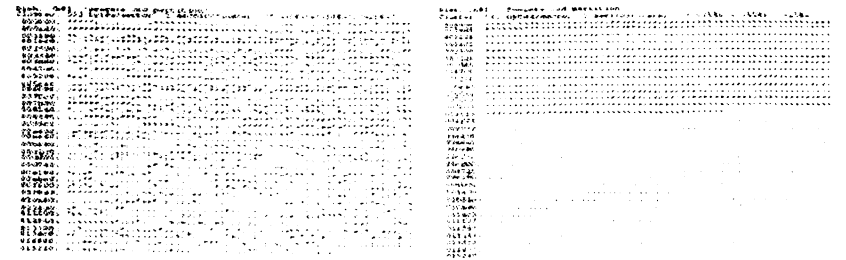
Subscriptions are **\$23/year for 8 issues** (every 6 weeks), or **\$12 for a 4 issue trial subscription** (\$30/\$16 for Canada, \$33/\$17 overseas). A disk service, "microdisk", is **\$40/year or \$6 per issue** (\$44/\$7 Canada, \$54/\$8 overseas). **First issue will be delivered in August... DON'T MISS IT!**

**FARNA Systems PB**  
 P.O. Box 321  
 Warner Robins, GA 31099-0321  
 Phone 912-328-7859

# Defragment your OS-9 disk with **Disk Squeezer.™**

Frequent writings of small records of different files result in file contents scattered on the disk; this is called "Disk Fragmentation." A fragmented disk usually takes longer access times because the disk heads have to travel back and forth on the disk surface to read and write logically contiguous but physically fragmented file contents. Even worst, if a file is fragmented into too many pieces (48 for 256 byte sectors), you may not be able to extend the file size by even a single byte even though the disk has plenty of free space. This is called "File Fragmentation" and is fundamental problem with the RBF file manager.

Disk Squeezer automatically detects fragmented files and makes them more contiguous. Squeezer also recognizes disk contents so that free sectors will be concatenated contiguous. This prevents future file fragmentation as well as disk access performance degradation. See the "before" and "after" graphical sector usage analysis of a real hard disk below.



Before Squeezing

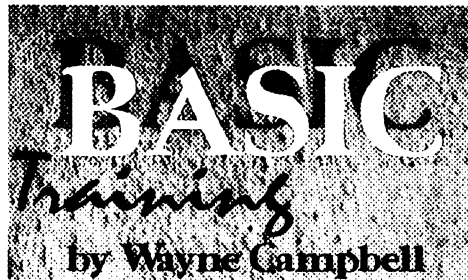


After Squeezing  
**Disk Squeezer: \$295.00**

- Other OS-9/68K programs available from ARK Systems USA:
- UD-Cache II - Light Speed Disk Cache** \$149.00
  - LSrcDbg - Split Displays for Debugger and Application** \$50.00
  - XSCF - Enhanced Line-Editing and Line-Recalling File Manager** \$60.00
  - DDF - Dev Device File Manager** *Coming soon*
  - PTF - Pseudo Terminal File Manager** *Coming soon*

\*All programs work on any OS-9/680x0 system (V2.2-2.4). Fragmentation improvement factor may vary.  
 S&H: US (48 states) orders add \$4.00 for ground service or \$11.50 for FedEx 2nd day air; AK, HI and outside US ask for quotation. CA residents add 8.25%. Send your check or money order (no charge cards or CODs) with preferred disk format (important). 10% discounts for OS-9 User's Group members; send copy of your membership card.





**NCode** **DCode**  
**Binary File Translators**

**S**ome of you may remember my ConvB routine from Volume 1 Issue 9. It created a Basic09 routine called ConvD that included the binary values of the file specified in DATA statements that would be converted back into binary form when it was run. That routine's main flaw was a size limitation. There is a certain size that you could not go beyond due to the memory limitations of Basic09. Well, I've written a new set of routines that allows you to convert any size binary file to a text file, and convert the text file back to binary code. They are called NCode (for eNcode) and DCode (for Decode).

These routines are relatively fast and should be useful for those who wish to transmit binary data and have no way to transmit the data directly as binary code. There are a few caveats however, and the remainder of this article will be devoted to them.

**NCode**

NCode is passed a single parameter; the pathlist of the binary file to be converted to text. The file may be anywhere on your disks (so long as the disk exists for floppy use). When run, any pathlist is removed from the specification for the purposes of creating the output file, as this file is created in your current data directory. The filename portion of the specification is used for the filename to create, with extension '.ND' appended to it.

*Continued Page 20*

0147	ENDIF	01C4	PUT #outpath,char
0149	UNTIL SUBSTR("#",outfile)=0	01CE	PRINT USING "h2",char;
0158	PRINT outfile	01DA	NEXT counter
015D	OPEN #inpath,filename:READ	01E5	PRINT
0169	inopen:=TRUE	01E7	ENDWHILE
016F	CREATE #outpath,outfile:WRITE	01EB 10	IF inopen THEN
017B	outopen:=TRUE	01F7	CLOSE #inpath
0181	WHILE NOT(EOF(#inpath)) DO	01FD	ENDIF
018C	READ #inpath,line	01FF	IF outopen THEN
0196	FOR counter:=1 TO LEN(line)-1	0208	CLOSE #outpath
	STEP 2	020E	ENDIF
01B0	char:=VAL("\$"+MID\$(line		
	,counter,2))		



**MultiBoot by Terry Todd & Allen Huffman**  
*Now have up to SIXTEEN bootfiles on your startup disk!*  
 Hot of the assemblers and compilers is a great must-have utility which lets you have up to 16 bootfiles on one disk! No more boot disk floppy-swapping! MultiBoot will install itself to a cobbled boot disk and, upon typing "DOS", will greet you with a scrolling menu of available bootfiles!  
**OS-9 Req: CoCo 3, OS-9 Level 2.....\$19.95**

Towel by Allen C. Huffman  
*The first EthaWin program - a disk utility for OS-9.*  
 A program no intergalactic hitchhiker should be without! Use a mouse or keyboard hot-keys to perform common file and disk commands from pull-down menus. Tag multiple files for Delete, Copy, Rename, etc., and even have point 'n click disk Backup, Cobbler, Dcheck and other commands. User menu lets you specify up to seven of your own commands to execute. Runs under the EthaWin interface on a high-speed text screen. All commands/colors configurable.  
**OS-9 Req: CoCo 3, OS-9 Level 2.....\$19.95**  
**OS/K Req: MM/1 or K-Windows Compatible.....\$24.95**

**992 CoCoFest SIMULATOR by Allen C. Huffman**  
*Graphics "adventure" based on the 1992 Atlanta CoCoFest*  
 The next best thing to having been there! Digitized graphics of the event and a text command parser (ie, "get the box of disks") let you see all the vendors and even run into some famous faces of the CoCo Community. The show area, seminar room, and portions of the hotel are all represented. No true "goal", but you do have to figure some things out, like how to get into the show and how to buy items from vendors. Runs on a 640x192 hi-res graphics screen.  
**OS-9 Req: 512K CoCo 3, OS-9 Level 2, 490K Disk Space.....\$ 9.95**  
**OS/K Req: MM/1 or 100% K-Windows Compatible.....\$14.95**

Send US funds plus \$2.50 shipping to:  
**Sub-Etha Software**  
 P.O. Box 152442  
 Lufkin, TX 75915  
**(815) 748-6638**  
 More items available! Contact us for a complete product listing!

Continued from Page 20

PROCEDURE NCode

```

0000 PARAM infile:STRING[255]
000C DIM inpath,outpath,char,counter
      :BYTE
001F DIM inopen,outopen:BOOLEAN
002A DIM outfile:STRING[255]
0036 inopen:=FALSE
003C outopen:=FALSE
0042 ON ERROR GOTO 10
0048 counter:=0
004F outfile:=infile+".ND"
005D REPEAT
005F IF SUBSTR("/",outfile)>0 THEN
006F   outfile:=MID$(outfile,SUBSTR
      ("/",outfile)+1,LEN(outfile))
0086 ENDIF
0088 UNTIL SUBSTR("/",outfile)=0
0097 PRINT outfile
009C OPEN #inpath,infile:READ
00A8 inopen:=TRUE
00AE CREATE #outpath,outfile:WRITE
00BA outopen:=TRUE
00C0 WHILE NOT(EOF(#inpath)) DO
00CB GET #inpath,char
00D5 counter:=counter+1
00E0 PRINT #outpath USING "h2",char;
00F0 PRINT USING "h2",char;
00FC IF counter=22 THEN
0108 PRINT #outpath
010E PRINT
0110 counter:=0
0117 ENDIF
0119 ENDWHILE
011D IF counter>0 THEN
0129 PRINT #outpath
012F PRINT
0131 ENDIF
0133 10 IF inopen THEN
    
```

```

013F CLOSE #inpath
0145 ENDIF
0147 IF outopen THEN
0150 CLOSE #outpath
0156 ENDIF
0158 END
    
```

PROCEDURE DCode

```

0000 PARAM infile:STRING[255]
000C DIM inpath,outpath,char:BYTE
001B DIM counter:INTEGER
0022 DIM ext:STRING[3]
002E DIM line:STRING[44]
003A DIM filename,outfile:STRING[255]
004A DIM inopen,outopen:BOOLEAN
0055 inopen:=FALSE
005B outopen:=FALSE
0061 ON ERROR GOTO 10
0067 ext:=RIGHT$(infile,3)
0072 FOR counter:=1 TO 3
0082 IF MID$(ext,counter,1)="n" OR
      MID$(ext,counter,1)="d" THEN
00A3   ext:=LEFT$(ext,counter-1)
      +CHR$(ASC(MID$(ext,counter,1)
      )-32)+RIGHT$(ext,LEN(ext)
      -counter)
00CE ENDIF
00D0 NEXT counter
00DB IF ext=".ND" THEN
00EA   outfile:=LEFT$(infile,LEN
      (infile)-3)
00FA   filename:=infile
0102 ELSE
0106   outfile:=infile
010E   filename:=infile+".ND"
011C ENDIF
011E REPEAT
0120 IF SUBSTR("/",outfile)>0 THEN
0130   outfile:=MID$(outfile,SUBSTR
      ("/",outfile)+1,LEN(outfile))
    
```

**AniMajik®**  
Productions

4650 CAHUENGA BLVD., STE #7  
TOLUCA LAKE, CA 91602  
(818) 761-4135

CHECK OR M.O. ACCEPTED

# Happy New Year!

*New Version!*

**DCom v3.5 - BASIC09 Decompiler ... \$24.95**  
Upgrade from previous version .. \$10.00

**C MicroCharts - C Syntax at your finger-tips. 8 1/2 x 11 Stain-resistant Plastic .....\$5.00**

**OSK ToolKit - A collection of useful utilities (by Leonard Cassady, C Software Engineering) for your OSK box .....\$9.95**

## Moving?

If you are planning to change your address, you must let us know at least 30 days in advance in writing. Send a postcard or letter to **Fat Cat Publications** with your current address and your new address so we can update your records and keep your subscription uninterrupted. *We are not responsible for your subscription, if we are not notified in time.*

**Fat Cat Publications**  
4650 Cahuenga Blvd., Ste #7  
Toluca Lake, CA 91602

## 68XXX COMPUTER PRODUCTS from Peripheral Technology - a company with a reputation for quality!

PT68K4-16, 1MB \$299.00  
PT68K2-10, 1MB \$199.00  
ALT86 for PC  
Compatibility \$199.00  
Profess. OS9 \$299.00

1480 Terrell Mill Rd. #870  
Marietta, GA 30067  
404/973-2156

As an example:

```
ncode /dd/sys/stdfonts
```

will look for the file stdfonts in the /dd/sys directory and will create a file named stdfonts.ND in the current data directory.

Also:

```
ncode stdfonts
```

will look for the file stdfonts in the current data directory and will create a file named stdfonts.ND in the current data directory. If the file specified does not exist, NCode will end *without* reporting the error.

The file created will contain strings of text that are 44 characters in length (22 bytes of data translated into text). The value of each byte of binary code is converted to text form in hexadecimal format. This makes for a uniform file that is easily transmitted elsewhere, and is easily converted back into binary code. If the last line of the file contains less than 22 bytes worth of data, it will be less than 44 characters in length.

### DCode

DCode is passed a single parameter; the pathlist of the text file to be converted to binary. The file may be anywhere on your disks (so long as the disk exists for floppy users). When run, any pathlist is removed from the specification for the purposes of creating the output file, as this file is created in your current data directory. The filename portion of the specification is used for the filename to create. If the extension '.ND' is not present (upper or lower case) it is appended to the filename to find.

As an example:

```
dcode /dd/binaries/stdfonts
```

will look for a file named stdfonts.ND in the /dd/binaries directory and will create a file named stdfonts in the current data directory.

Also:

```
dcode stdfonts.nd
```

will look for a file named stdfonts.nd in the current data directory and will create a file named stdfonts in the current data directory. If the file specified does not exist, DCode will end *without* reporting the error. The file created will be a duplicate of the original binary file.

Additionally, these routines may be used to translate executable files. The file created by DCode will also be executable, but the attributes of the file will not be set for execution. You'll have to set the attributes to be able to execute the file.

NCode and DCode *do not* report errors. You'll want to check the files created when NCode or DCode are run to verify their integrity. These routines may be altered to remedy this, if you so choose, but I would like all changes, additions or alterations to be reported to me via this magazine, or through email to me. My email address is:

zog!wayne@abode.ttank.com  
(or use one of the addresses on page 15)

Enjoy!

*Continued Page 22*

**Advertise in the  
OS9 Underground**  
Rates start as low as \$7.50 (¼ Page ad)  
Call for Rate Card or Information  
**(818) 761-4135**

# Northern Xposure

'Quality Products from North of the Border'

**Smash!** **\$29.95**  
Breakout-style Arcade Game for OS-9 Level II

- 1 or 2 player mode
- 32 levels, can be edited
- 17 different block types
- multiple balls
- 320x192x 16 color graphics
- written by Alan DeKok, author of CC3Demo and Thexder:OS-9

*Many other fine products available  
Write for a free catalogue*

- Prices are US funds and include S&H ●

Alan DeKok

7 Greenboro Cres  
Ottawa, ON  
Canada K1T 1W6  
(613)736-0329

Colin McKay