

72

COMPUTIST

Canada & Mexico \$7

U.S. \$3.75

From Richard Brooks: The best price that I have found on *Beneath Apple DOS* and *Beneath Apple ProDOS* is from Silicon Express (1-614-927-9555) for \$12.95 each plus approximately \$4.00 for shipping, depending on where you live.

&

The *new* BBS NEWS Column

&

A fat issue to help with those Softkey hunger pangs!

**Subscribe to
COMPUTIST
We give
you More!**

**Only \$24 for your
first 8 issues**

I am:

- A new subscriber
- Renewing my current subscription
- Changing my address (Please include last mail label)

Subscription rates:

- U.S. (bulk)\$24
- U.S. First Class / Canada / Mexico\$34
- All other Foreign\$54

Send to:

**COMPUTIST
33821 E Orville Rd
Eatonville WA 98328
(206) 474-5750**

Name _____ ID# _____

Address _____

City _____ State _____ Zip _____

Country _____ Phone _____

Visa MC _____ Exp. _____

Signature _____ CP72

•Send US funds drawn on US bank. • Please allow 4-8 weeks for 1st issue or add \$1.25 for postage and we will send your 1st issue by 1st Class mail. • Send check/money order to:

COMPUTIST 33821 E Orville Rd Eatonville WA 98328 (206) 474-5750

**COMPUTIST
33821 E Orville Rd
Eatonville WA 98328**

**BULK RATE
U.S Postage
PAID
Tacoma WA
Permit No. 269**

COMPUTIST

Charles R. Haight Editor
Tom Acree Associate Editor
Karen Fitzpatrick Circulation
Jeff Hurlburt Reviews

COMPUTIST is published by SoftKey Publishing. Address all advertising inquiries to:

COMPUTIST
Advertising Department
33821 East Orville Road
Eatonville, WA 98328
(206) 474-5750

Mail all RDEX letters to:

COMPUTIST
Apple-RDEX or IBM-RDEX
33821 East Orville Road
Eatonville, WA 98328

• COMPUTIST does NOT purchase editorial material. The entire editorial content consists of information submitted to COMPUTIST for publication in the shared interests of all COMPUTISTS.

• Unsolicited material (manuscripts, letters to the editor, softkeys, A.P.T.s, playing tips, questions, etc.) are assumed to be submitted as letters-to-the-RDEX-editor for publication with all and exclusive rights belonging to COMPUTIST.

• Entire contents copyright 1990 by SoftKey Publishing. All rights reserved. Copying done for other than personal or internal reference (without express written permission from the publisher) is prohibited.

• The editorial staff assume no liability or responsibility for the products advertised in this newsletter. Any opinions expressed by the authors are not necessarily those of COMPUTIST magazine, its staff or SoftKey Publishing.

SUBSCRIPTIONS: Rates (for 8 issues):
U.S.\$24 Canada/Mexico\$34
U.S. 1st Class\$34 Other Foreign\$54

• Send subscription inquiries to:

COMPUTIST
Subscription Department
33821 East Orville Road
Eatonville, WA 98328

• Domestic Dealer rates: Call (206) 474-5750 for more information.

• Change Of Address: Please allow 4 weeks for change of address to take effect. On postal form 3576 supply your new address and your most recent address label. When we receive your notice of change of address, we will send you a acknowledgement card. If you do not receive the acknowledgement card after 2 weeks, send another notice or call us direct.

Issues missed due to non-receipt of change of address may be acquired at the regular back issue rate.

We are not responsible for missing issues 90 days after mailing date. If you do not receive an issue at the usual time each month, please call or write.

Apple® is a trademark of Apple Computers. IBM® is the IBM trademark.

Readers Data EXchange

New COMPUTIST readers using Apple II's are advised to read this page carefully to avoid frustration when attempting to follow a softkey or entering the programs printed in this issue.

What is a softkey, anyway?

Softkey is a term which we coined to describe a procedure that removes, or at least circumvents, any copy-protection on a particular disk. Once a softkey procedure has been performed, the resulting backup copy can usually be copied by the normal copy programs (for example: COPYA, on the DOS 3.3 System Master disk).

Commands and control keys

Commands which a reader is required to perform are set apart by being in boldface and on a separate line. The return key must be pressed at the end of every such command unless otherwise specified. Control characters are preceded by "ctrl". An example of both is:
6 ctrl P

Type 6. Next, place one finger on the ctrl key and then press P. Don't forget to press the return key.

Other special combination keypresses include ctrl reset and open-apple ctrl reset. In the former, press and hold down the ctrl key then press the reset key. In the latter, press and hold down both ctrl and open-apple then press reset.

Software recommendations

The Starter Kit contains most of the programs

that you need to "Get started". In addition, we recommend that you acquire the following:

- Applesoft program editor such as "Global Program Line Editor (GPLe)".
- Assembler such as "Merlin/Big Mac".
- Bit-copy program such as "Copy II Plus", "Locksmith" or "Essential Data Duplicator".
- Word-processor (such as AppleWorks).
- "COPYA", "FID" and "MUFFIN" from the DOS 3.3 System Master disk.

Super IOB and Controllers

This powerful deprotection utility (in the COMPUTIST Starter Kit) and its various Controllers are used in many softkeys. (It is also on each Super IOB Collection disk.)

Reset into the Monitor

Softkeys occasionally require the user to stop the execution of a copy-protected program and directly enter the Apple's system monitor. Check the following list to see what hardware you will need to obtain this ability.

Laser 128: Your ROM includes a forced jump to the monitor. Press ctrl return reset.

Apple II+, //e, compatibles: 1) Place an Integer BASIC ROM card in one of the Apple slots. 2) Use a non-maskable interrupt (NMI) card such as Replay or Wildcard.

Apple II+, compatibles: 1) Install an F8 ROM with a modified reset-vector on the computer's motherboard as detailed in the "Modified ROM's" article (COMPUTIST #6 or Book Of Softkeys III) or the "Dual ROM's" article (COMPUTIST #19).

Apple //e, //c: Install a modified CD ROM on the computer's motherboard that changes the open-apple ctrl reset vector to point to the monitor. (This will void an Apple //c warranty since you must open the case to install it.)

Apple //gs: If you have the 2.x ROM, there is a hidden Classic Desk Accessory (CDA) that allows you to enter the monitor. In order to install the new CDA, you should enter the monitor (CALL -151) before running any protected programs and press # return. This will turn on two hidden CDAs, Memory Peeker and Visit Monitor. Thereafter press open-apple ctrl esc to go to the Desk Accessories menu. Select Visit Monitor and there you are. Use ctrl Y to exit.

Recommended literature

- Apple II Reference Manual (or IIe, IIc, etc.)
- DOS 3.3 or ProDOS manual
- Beneath Apple DOS & Beneath Apple ProDOS, by Don Worth and Pieter Lechner, from Quality Software

Typing Applesoft programs

BASIC programs are printed in a format that is designed to minimize errors for readers who key in these programs. If you type:

```
10HOME:REMCLEAR SCREEN
```

The LIST will look like:

```
10 HOME : REM CLEAR SCREEN
```

Applesoft inserts spaces into a program listing before and after every command word or mathematical operator. These spaces don't pose a problem except when they are inside of quotes or after a DATA command. There are two types of spaces: those that have to be keyed and those that don't. Spaces that must be typed appear in COMPUTIST as special characters (◊). All other spaces are there for easier reading.

NOTE: If you want your checksums to match, only type spaces within quotes or after DATA statements if they are shown as (◊) characters. SAVE the program at periodic intervals using the name given in the article. All characters after a REM are not checked by the checksum program so typing them is optional.

Typing Hexdumps

Machine language programs are printed in COMPUTIST as hexdumps, sometimes also as source code.

Hexdumps are the shortest and easiest format to type in. You must first enter the monitor:
CALL -151

Key in the hexdump exactly as it appears in the magazine, ignoring the four-digit checksum (\$ and four digits) at the end of each line. When finished, return to BASIC with:
3D0G

BSAVE the program with the filename, address and length parameters given in the article.

Typing Source Code

The source code is printed to help explain a program's operation. To enter it, you need an "Assembler". Most of the source code in older issues is in S-C Assembler format. If you use a different assembler, you will have to translate portions of the source code into something your

assembler will understand.

Computing checksums

Checksums are 4-digit hexadecimal numbers which tell if you typed a program correctly and help you locate any errors. There are two types of checksums: one created by the CHECKBIN program (for machine language programs) and the other created by the CHECKSOFT program (for BASIC programs). Both are on the "Starter Kit".

If your checksums do not match the published checksums then the line where the first checksum differs is incorrect.

CHECKSOFT instructions: Install Checksoft (BRUN CHECKSOFT) then LOAD your program. Press & to get the checksums. Correct the program line where the checksums first differ.

CHECKBIN instructions: Enter the monitor (CALL -151), install Checkbin at some out of the way place (BRUN CHECKBIN, AS6000), and then LOAD your program. Get the checksums by typing the Starting address, a period and the Ending address of the file followed by a ctrl Y.

SSSS.EEEE ctrl Y

Correct the lines where the checksums differ.

Writing to the RDEX editor

RDEX (are-decks) stands for: Reader's Data EXchange. We print what you write. When you send in articles, softkeys, APTs, etc., you are submitting them for free publication in this magazine. RDEX does not purchase submissions nor do we verify data submitted by readers. If you discover any errors, please let us know so that we may inform our other readers.

Remember that your letters or parts of them may be used in RDEX even if not addressed to the RDEX editor. Correspondence that gets published may be edited for clarity, grammar and space requirements.

Because of the great number of letters we receive and the ephemeral and unpredictable appearance of our volunteer staff, any response to your queries will appear only in RDEX, so it would be more appropriate for you to present technical questions to the readers and ask for their responses which will then be placed in the Apple-RDEX.

How to get a free library disk

Whenever possible, send everything on Apple format (5.25" - DOS/ProDOS or 3.5" - ProDOS) or IBM format (3.5") disks. Other formats are acceptable but there may be some delay as we look for someone to translate it for us. (If you use a 5.25" disk, when we print your letter, we will return your disk with the current library disk copied onto it.) Use whatever text editor you like, but tell us which one. Put a label on the disk with your name (or pseudonym) and address (if you want to receive mail). Don't reformat any programs or include them in the text of your letter. Send Applesoft programs as normal Applesoft files and machine language programs as normal

binary files. We have programs to convert them to the proper format for printing. If you are sending source code files, and you are not using the S-C Assembler, send them as normal text files.

When to include a printed letter

Don't include hardcopy (printout) unless:

- a. You are writing about a bug or other printing error.
- b. You are writing to ask for help.
- c. You are answering another readers help request.
- d. You are writing about your subscription or sending an order for back issues or software.

Bugs, requests for help and answers to requests for help are bumped to the head of the line and go in the very next issue. All other letters are printed in the order that we receive them.

Writing to get help

When writing to request help, be sure to include ALL relevant information. The more information you include, the easier it is to find a solution. There's an old saying that goes "A properly framed question includes 90% of the answer".

How to get mail

If you are interested in receiving mail from other readers, be sure that we have a current address. If you use a pen name and want to receive mail, we need to have your address. Our readers privacy is important, so we will not print your address unless you specifically say too.

How to write to RDEX authors

When writing to one of the RDEX authors. Write your letter and seal it in an envelope. Put your return address, the authors name (as it appears in RDEX) and the correct postage on the envelope. Put this envelope into another and send it to RDEX. We will put the correct address on your letter and mail it for you.

Help Line

These readers have volunteered their time to help you. Please call only within the given time frames (corrected for your time zone). No collect calls.

Jack Nissel (Disk Protection, 7-10PM EST)
.....(215) 365-8160

Electronic Bulletin Board System (BBS)

Dave Goforth is the sysop for the Computist BBS. The number is: (206) 581-9292. If you already have a User ID# and password, sign-on using the User ID#. If you are a new user, it may take a day or so to validate your new ID# and password.

You have a LEGAL RIGHT to an unlocked backup copy of your commercial software.

Our editorial policy is that we do NOT condone software piracy, but we do believe that users are entitled to backup commercial disks they have purchased.

In addition to the security of a backup disk, the removal of copy-protection gives the user the option of modifying programs to meet his or her needs.

Furthermore, the copyright laws guarantee your right to such a DEPROTECTED backup copy:

... "It is not an infringement for the owner of a copy of a computer program to make or authorize the making of another copy or adaptation of that computer program provided:

1) that such a new copy or adaptation is created as an essential step in the utilization of the computer program in conjunction with a machine and that it is used in no other manner, or

2) that such new copy or adaptation is for archival purposes only and that all archival copies are destroyed in the event that continued possession of the computer program should cease to be rightful.

Any exact copies prepared in accordance with the provisions of this section may be leased, sold, or otherwise transferred, along with the copy from which such copies were prepared, only as part of the lease, sale, or other transfer of all rights in the program. Adaptations so prepared may be transferred only with the authorization of the copyright owner."

United States Code title 17, §117

Table of Contents

Editorial Note	3
The Product Monitor	7
2088: The Cryllan Mission	7
Bad Duds	8
Cartooners	7
Copy II Plus, 9.0	8
Printing Out the Parm Names	9
Correction: Bums Don't Star	8
Crusin' on Crylla: 2088	8
Geometry	7
Gnarly Golf	7
Great Western Shootout	8
Laser Force	8
Matching Madness	8
No Foolin'	9
Postcards	7
Skate or Die	9
The Legend Of Blacksilver	8
Uninvited	9
Vendors	9
Warlock	7
Workers Wanted (Dark Castle)	8
RDEX Contributors	39
Most Wanted Softkeys	39
Softkeys:	
Addition & Subtraction	36
Aesop's Fables IIe	33
Algebra 1	36
Algebra 2	36
Algebra 3	36
Alphabet	36
An Introduction to General Chemistry	9
APBA Major League Players Baseball	37
Arkanoid	23
Arkanoid II: The Revenge of DOH (GS)	20
Balance	36
Balance of Power 1990 v. 2.08	28

Biosolve	14
Bubble Ghost (GS)	20
Building memory Skills	15
Calendar Crafter v1.2	26
CBS's Pathwords	36
Charlie Brown's 1,2,3's	36
Chessmaster 2100 v1.1	28
Children's Writing & Publishing	
Center	20, 22, 37
Conquering Decimals * and /	27
Conquering Decimals + and -	27
Conquering Fractions; * and /	27
Conquering Fractions; + and -	27
Coordinate Math	27
D.C.Heath/MB	36
Dark Lord	23
DataEast Games	18
Decimal Concepts	27
Decimal Discovery	36
Decimals	36
Delta Drawing 3.33	15
Designasaurus	14
Dig Dug	10
Duel (gs)	20
Early Games	36
Eliminator	25
Equation Math	27
Esimation	36
Explora-Classic series	14
Explora-Science Whales	14
Fraction Concepts	27
Fraction Practice Unlimited	27
Fraction Recognition	36
Fractions	36
Freddy's Puzzling Adventures	24, 36
Frogger	12
Galaxy Math Games	36
Galaxy Search	36
Garfield Companion	14
Garfield Trivia	14
Geometry (GS)	37
Ghostbusters	22
Gnarly Golf (GS)	29
Gnee or Not Gnee	26
Grammar Examiner	19
Graphics Studio	10
Guinness World Records	36
Homeworker	19
Houghton Mifflin Math Courseware	15
Ice Demons	23
Ivitation to Math series	36
Kid Niki	18
King of Chicago	29
Kittens, Kids, and a Frog	33
L.A. Crackdown	18, 28
Language Carnival	14
Magic Slate II	19
Magic Word	36
Master Match	14
Math And Me	19
Math Blaster	20
Math Blaster Plus	19
Math for Everyday Living	36
Math in a Nutshell	36
Math Masters	14
Math Tutor (Percents)	36
Mathematics Skills	18
Mavis Beacon Teaches Typing (IIe)	28
McGraw Hill Compucat Quizware	15
Memory Castle	37
Meteor Multiplication	36
Microzine #3	36
Microzine #8	36
Microzine #9	36
Microzine #10	37
Microzine #11	36
Microzine #12	37
Microzine #13	37
Microzine #15	37
Microzine #16	37
Microzine #22	37
Microzine #23	14
Microzines and Microzine Jr.	36
Mixed Numbers	36
Moptown Hotel	26
Mr. and Mrs. Patatohead	36
Mr. Pixel's Programming Paint Set	24
Mystery Matter	26
Mystery Objects	26, 27
Numbers	36
Opposite	36
Organic Chemistry	9
Pacman	30
Peanuts Math Matcher	36
Percentages	36
Perplexing Puzzles	24
Pick the Numbers	36
Picture Perfect	15
Pirates! GS	23
Pixelwerks	24
Platoon	23, 27
Police Quest	18
Pool of Radiance	25
Quest Strategy Checker	14
Rainbow Painter	19
Ratios & Propotions	36
Read 'N Roll 1.1	20
Read-Write-Publish	14
Reading Comprehension	36
Robomath	20
Sailing Through Story Problems	14
Scuffy and Friends	33
Sensible Speller (ProDOS)	18
Sierra Programs	23
Silicon Dreams	16
SOKO-BAN	28
Solving Quadratic Equations	20
Spelling Bee	36
Spelltronics	36
Spy's Adventure in South America	14
Square Pairs	36
Star Trek	29
Stickers	19
Study for Success	15
Success With Typing	14
Super Print	36
Tales of Fantasy	14
Talking Stickybear Opposites	24
Talking Text Writer	14
Teasers by Tobbs	37
Test Drive II: The Duel (Iigs)	20, 37
Test Taking Made Easy	15
The Boars Store *	36
The Boars Tell Time	36
The Duel: Test Drive II	36
The Game Show	14
The Games: Winter Edition	36
The Wonderful World of Paws	35
Time Capsule	14
Times of Lore	25
Townbuilder	36
Transylvania	15
VCR Companion	24
Voyage of the Mimi	19
What makes a Dinosaur sore	37
Where in the USA is	
Carmen SanDiego	28, 30
Where in the World is	
Carmen SanDiego? (GS)	19, 27, 30, 37
Who, What, Where, When, Why	26
Whole Number Operations	36
WISCR Intelligence Test	15
Wood Car Rally	26, 27
Word Attack Plus! Spanish	37
World Geograph	27, 28, 36
World History Adventure	33
Xenocide (GS)	20, 27
Bitkeys:	
Borg	25
Great Western Shootout	34
Laser Force	34
Math Blaster Mystery	20
The Hunt for Red October	27
APTs:	
Anti Gravity	25
Autoduel	16
Bard's Tale II	18
Chrono Warrior	25
Commando	18
Demonic Decks	25
Neuromancer	31
Pool of Radiance	25
Tetris	36
The Bards Tale III	23
Track & Field	17
Victory Road	23
Xevious	23
Zany Golf	31
Playing Tips:	
Bard's Tale II	18
Bard's Tale III	19, 28, 35
Defender	19
Defender Of The Crown (IIGS)	19
Leisure Suit Larry	19, 28
Moebius	15
Pool of Radiance	34
Print Shop Companion	19
Robocop	37
Tower of Myraglen	15
Wasteland	35
Who Framed Roger Rabbit	23
Wings Of Fury	19
Features, Notes and such:	
A Bug in Prentice Hall Science Courseware	15
A note on Stickybear encrypted sector	15
An accelerated IIe & EDD 4	15
Finding the licencee's name in GEOS	15
A BUG in Teacher's Tool Kit Series	19
A reader review of the Trac Card	10
An Explanation of Self-sync Bytes	13
Another reason why Cookbook Cracks	
might not work	33
BBS News	4
Feedback (To SysOp)	5
General Messages	4
Time Saver News	5

Beginners Guide to "PACMAN"	
deprotection	30
Bogus 18 sector disks?	25
Bug in Type! softkey	37
Bugs in Pool of Radiance	35
Changing levels on Tetris (IIe)	9
Cheats, Hints, and Tips for Neuromancer	31
Comments & possible help to	
other readers	33
Comments on IBM Ragging	10
Converting Print Magic Graphics	
to Publish It!	12
Copy Protecting Your Own Disks	
with ProDOS	34
Determining when individual files can be	
removed from a Protected Disk	33
DOS 3.3P (for protected)	25
Enhancing DOS 3.3	17
FIND.CAT Enhancement	24
Half & quarter Tracks	25
Installing GS/OS on Sierra 3D Adventures	26
Lifting the Lid on COPYA	11
Making Jack Nicklaus Greatest 18 Holes	
of Championship Golf play faster	12
Moving Destroyer GS (Epyx)	
to Hard Disk	35
Note on '84 & '85 MECC disks	14
Notes on programs published	
by Micrograms	33
Notes on Time Out and Appleworks 2.1	36
Notes on Ultima IV and V	35
Print in Color with Appleworks	34
Put Autoduel on a 3.5" diskette	15
Reading From Protected ProDOS Disks	17
Removing the Manual Check from	
Pirates! GS	23
Running other programs	
from your Hard Disk	36
Super 6.0 FastcopyA	20
Fun with Super 6.0 FastcopyA	22
Turn Dig Dug into a BRUNable File	10
Ultimapper V	24
IBM Softkeys:	
688 Attack Sub	38
Battle Chess	38
Battle Hawks 1942	38
Chuck Yeager's Advanced Flight	
Simulator v1.0	38
Shinobi	38
IBM Feature & Notes:	
Bug in APT for Zany Golf	38
Cracking on the IBMpc	38

Editorial note

Catching Up with Submissions?

That's right! This large issue marks the beginning of a concerted push to catch up with submissions.

As of the next issue (#73) we will be current with the March submissions which will put us back into "real time" printing. And that means everything will be going into the next issue instead of waiting for several months.

That also means that we will be able to tell you what issue your article is going to be printed in.

If you don't see your material in the next issue, call or write and let us know. It may be one of the articles that requires extensive editing and got bumped to the next issue.



There are currently over 1,040 BBS members and another 50 or so subscribers on the board. Thanks to Ed Croft, we now have a clock installed on the BBS. Another subscriber has donated a registered version of GBBS (which numerous callers have recommended.) As Mr. Haight stated, there will be a lot of changes going on in the beginning (many have already occurred). As of now (24 Mar), I've changed the logon routine for faster logon & it will show the user# that you are about to logon as so, if you make a mistake & type the wrong user# or receive line noise you can press return at the password prompt and reenter your user#. You may also list users starting at a specific user# if you wish (in case you remember Joe is user# 600 & something, you don't have to go through the first 599 users). The Computist index is available for downloading in the "EDITORIAL" file base as well as an on-line index in the "On-Line" area. The file base has two versions, one is in Appleworks data base (ADB) format and the other is a standard ASCII (TXT) text file. The ADB version is shrunk with ShrinkIt and will fit on a single 5.25" disk (140K). The text version is not archived and requires at least a 3.5" disk. Both contain the same information with the exception that one may be slightly newer than the other as indicated by the extension (.##) which indicates that it is complete thru issue ##. This index is available courtesy of J.L. Walters.

General Messages

Welcome all new users to the Computist BBS.

This message base is designed for all public messages between users on this system. It is accessible by everyone and the only restriction to it's use is that I ask that you keep conversations and language in good taste. The only other thing I ask is that you do NOT use this area or this BBS as a means to discuss or setup 'PIRATE' meetings... This BBS is based on the principle that 'All owners of legally purchased software have a right to make backups of that software.' and not for the purpose of redistributing it to others. See the bulletin 'PIRACY & us' for more info on this...

Again, welcome to the BBS & enjoy...
<<<SYSOP>>>

Note #2

Date: (No Clock Installed)
To: ALL
From: (User #226)
Subj: THE MOST WANTED LIST

The most wanted list seems to be forever growing... The problem as I see it is that 18 sector programs abound.. (The new Electronic Arts for instance).. I was compiling a list of the software on the list and what protection it uses.. Thought that this would be of some help to the teeming millions out there... I can't help soft, half, bit or hard key the disks since I either no longer have an original or a copy of the programs..
Edward Teach

Note #5

Date: (No Clock Installed)
To: ALL
From: (User #543)
Subj: HOT TIP

Hey I don't know if this works or what it does....(since I don't have the software)....but in Cavern Cobra hit Ctrl-R. If this really does anything let me know what eh???

CTRL-R starts the game Cavern Cobra...
<sysop>

Note #8

Date: (No Clock Installed)
To: ALL
From: (User #531)
Subj: MATHBLASTER PLUS IIE

Does anyone out there know how to crack Mathblaster Plus for the Iie on a 3 1/2 disk? I have found cracks for both the IIGs version and for the 5 1/4 inch version on Compuserve. If you

know, please leave me a message. My user ID is 531. Thanks.

Note #9

Date: (No Clock Installed)
To: SYSOP
From: (User #1051)
Subj: NEW MEMBERSHIP

I RECEIVED THE COMPUTIST TODAY AND SAW ON THE FRONT PAGE ABOUT YOUR BBS, HAD TO CALL AND FIND OUT WHAT ITS ALL ABOUT, AND TO WISH YOU LUCK IN YOUR VENTURE. ALSO WOULD LIKE TO TELL YOU NO SLIP OF PAPER WAS FOUND IN MY COPY, SO I WOULD LIKE TO LET YOU KNOW MY NUMBER ON MY SUBSCRIPTION LABEL IS #0215, IF THAT WILL HELP ANY ON VALIDATION TO THE BBS.
THANK, SCOTT

Just about everyone's label says 0215 (02 = Feb, 15 = day that the issue was mailed...) The slip of paper was only in the issues for those that paid the BBS membership cost already (\$10 annual fee for full access)...<sysop>

Note #10

Date: (No Clock Installed)
To: ALL
From: (User #622)
Subj: MIGHT & MAGIC II

DOES ANYONE KNOW HOW TO KRACK MIGHT MAGIC II ANY HELP WOULD BE APPRECIATED DISK A CAN BE COPIED BUT WHEN YOU TRY TO SAVE ANY CHANGES IT WILL NOT SAVE THEM. MY USER ID# IS 622 THANKS

Note #11

Date: (No Clock Installed)
To: ALL
From: (User #1052)
Subj: BUNCHA STUFF...

Heya y'all... well I love the magazine, and I love the board. So what's left? This message is to any assembly hackers out there... I'm interested in doing a 'LIFE' for the GS or //e, so if you could offer advice, etc. I'd sure love the input. Also, does anyone know about an implementation of LISP on the GS? It would take up hellish amounts of memory but I think it would be a worthy project. I'm still working on learning the Toolbox routines, maybe I'll attempt it later (with a little (read: lot) help from my friends). So that's about it for now... ciao,
Thankee, Dan W. AKA The Gray Lensman

Note #23 (Reply to #11)

Date: 4 Mar 90 (06:44)
From: (User #1062)
Subj: BUNCHA STUFF...

I know of two LISP compilers for the regular // series. Check out page #20 of Computist #71 for their addresses.
JsH

Note #12

Date: (No Clock Installed)
To: ALL
From: (User #837)
Subj: HI

Howdy all-
I have been a long-time subscriber to Hardcore Computist and have learned a lot from my complete set. I think this BBS will be a great enhancement. Any teachers reading this might want to call my BBS- THE COLONY (209)233-2937 1200 BAUD 8N1 24 hours. Lets help each other deprotect educational wares before the kids kill the originals.
Loogie Hacker from Clovis, CA

Note #22 (Reply to #6)

Date: 4 Mar 90 (06:42)
To: SYSOP
From: (User #1062)
Subj: BBS

Since this BBS is up and running now, why not offer a telecom disk to the readers. The disk would contain several packing/unpacking programs, as well as a telecom program or two for the regular // series and the IIGS. The telecom programs that come to mind are Talk is Cheap (older version) and Freeterm GS (a rotten program, but at least it is free).
JsH

Reply:

If I understand you correctly, you mean give this away for uploading articles or softkeys, etc. for publication. This may not be a bad idea however, as it stands, Computist is in no position to purchase such items for promotional enticement on a continuing basis. But, what I will do is this. From 1 April to 30 April, whoever sends in the most information for publication, I will personally donate as prizes the following:

1. Quicken (Checking & Finance software) for the Apple IIe/cigs (complete with documentation & registration card, never used)..
2. Freeware disk containing: ShrinkIt,

FredWriter, and other useful freeware utilities for both the Iieic and GS.

3. Five disks (3.5") containing SoundSmith (shareware) which is the newest music generator for the IIGs (4 disks are music files for the program) which requires GS/OS 5.02

Does this sound like a reasonable start? Whoever sends the best full featured article will receive their choice of the above. The person who sends the most information by volume will receive their choice from the other two. The third will carry over to the next contest. The submissions may be submitted through the BBS or mailed directly to Computist (remember, it must be received by 30 April if mailing)... Any comments concerning this are welcome (good, bad, or suggestions), I'm not richer than Computist but if it generates support then I'll do what I can... One more thing, I will personally give to the first person who can get 100 new subscribers (or talk a current subscriber into renewing) by Dec 31st 90 their choice of software or hardware up to \$100 in value. Be sure to tell them to say "YOU" recommended the magazine. Whoever get's this will surely have earned it & I will also throw in an additional item (appropriate to their machine) as a surprise bonus. The contest will officially end when a single person reaches the 100 mark, whether it be a week, month, or last day of 90.

<<< SYSOP >>>

Note #26

Date: 5 Mar 90 (16:55)
To: ALL
From: (User #1052)
Subj: PROTREE BBS

I have an old (pre-GS) version of ProTree's ALFA BBS system around here, with manuals. This system reminds me of it. Personally, I'm a long-time fan of GBBS but who knows if they're still around?

My nickel's worth, D.W. aka The Gray Lensman

Note #35 (Reply to #26)

Date: 10 Mar 90 (21:49)
From: (User #601)
Subj: PROTREE BBS

Being new to BBS'ing I can't talk from experience but all of the Apple boards I have seen until now are GBBS. R.G aka Dragon Master

Note #27

Date: 5 Mar 90 (17:41)
To: ALL
From: (User #357)
Subj: GREETINGS

I send greetings to all fellow Computist nut & fellow Hackers!

I'm so happy I can hardly contain myself. It's here! Long live computist and the computist BBS!

Now on to business.

I am seeking someone who has a copy of the "fastcopy" program from the old locksmith program that runs on the GS. I have tried to follow the article in the magazine but was repeatedly unsuccessful. (mark that up to clutch error) Anyway, if anyone out there has accomplished this feat I would like very much to speak to you. Please send me E-Mail or if convenient give me a call voice at (513)277-2400 - I'll accept a quick collect call just be sure and state the nature to the operator. Thanks for the help!!

Note #29

Date: 8 Mar 90 (20:03)
To: ALL
From: (User #1099)
Subj: WALT DISNEY SOFTKEY

DOES ANYONE KNOW OF A SOFTKEY FOR WALT DISNEY CARD AND PARTY SHOP AND WALT DISNEY COMIC STRIP MAKER. THERE IS ONE IN ISSUE #50 BUT IT DOESN'T WORK FOR MY DISKS. PLEASE HELP AS I HAVE LOST THE ORIGINALS THAT CAME WITH THE PROGRAMS AND I ONLY HAVE THE BACKUPS. MY BOX IS 1099.

Note #30

Date: 8 Mar 90 (21:58)
To: SYSOP
From: (User #1100)
Subj: COMPUTIST

I don't know who started up Computist but it is what I've been looking for, for a long time. (ever since I got my gs!) a year ago. I just got a copy of your mag. A friend of mine, who just happens to be floating around the world on the Carl Vinson, subscribes to your mag and he sent me the latest issue he got. Already it has been useful. I got Children's publishing center cracked thanks to your mag. Anyway's I'm looking forward to being validated.

Thanks,
Paul

Charles Haight started Computist in 1981 (I think) & is still running it today. Used to be doing much better but with everyone's help it can soon return to it's former glory... We're working on it

slowly but surely and in the right direction. Thanks to all of it's loyalists... <sysop>

Note #31

Date: 10 Mar 90 (14:09)
To: ALL
From: (User #60)
Subj: HELLLLLLLL

Hi I can't follow exactly the softkey for super print 3.5" vers as it is in the Computist. And, since my version is just a little different; I can't crack it.

Also, does anyone have a krack for Carmen Europe on the 3.5" disk

Thanks—send E-mail to Mike Paterno

Note #34

Date: 10 Mar 90 (21:31)
To: ALL
From: (User #543)
Subj: HARDWARE

hey people i don't know about you guys, but i'd like to see a nmi-card in the hardware corner for the gs. a few of the programs are shutting off the control panel and access to the monitor...a few rom routines wouldn't be bad either.... (or does anyone know how to get around this problem???)

It seems to me that the easiest way around this would be to find the code that disables the CDA access and disable it (since it's normally always there the code should be relatively close on all programs that do this - provided it's not encoded.) I'm not that familiar with my GS yet, how 'bout anyone else, any suggestions? <sysop>

Note #36

Date: 10 Mar 90 (21:55)
To: ALL
From: (User #601)
Subj: GNARLY GOLF

HELP!!! Does anyone know how to install Gnarly Golf on a hard drive, the manual says it is possible but when I put it on the hard drive it hangs up. I'm using GSOS on a CMS 60meg drive. Any Ideas? Dragon Master

Have you checked the manual? A lot of protected (guessing) programs require that the original still be in the drive for it's protection check routine. If it's not protected, some require certain path names for subdirectories. Another possibility is to look at the system folder (if using GS version) on the original & see if there is anything that is not in the system folder on the hard disk (usually in the TOOLS, DRIVERS, and SYSTEM SETUP folders), if so copy the missing piece(s) to the HD system folder...<sysop>

Note #56

Date: 17 Mar 90 (23:30)
To: SYSOP
From: (User #601)
Subj: GNARLY GOLF

It uses a key disk protection, when you run it from a 3.5 copy it will ask for the original; but from the hard drive it hangs up before that.

Sounds like it's using a hard encoded Volume name for the disk. To verify this try copying all files to the HD (just as they are on the 3.5, don't make a separate subdir) and rename the HD volume to the same name as the Gnarly Golf disk and try it (if it works just like the copy then the Volume name is hard encoded, you gotta find it & change it to accept any prefix). If this don't work, then their still doin' something else (hopefully, you have ensured that you have all the same / System/System.Setup/files on your HD as are on the disk as well as Fonts & Drivers)...<sysop>

Note #41

Date: 11 Mar 90 (11:07)
To: ALL
From: (User #547)
Subj: DUNGEON MASTER

I'M LOOKING FOR A SOFTKEY THAT WILL LET ME USE THE SAVE GAME OPTION.

Note #43

Date: 11 Mar 90 (18:31)
To: SYSOP
From: (User #1100)
Subj: COMPUTIST

I neglected to tell you that I will be sending in for subscription tomorrow 12 March. I can't wait for the back issues or for the new ones. I have been spreading the news about the pub down here (Antioch, the Bay area) and hope that we can keep it alive

Note #44

Date: 11 Mar 90 (20:28)
To: SYSOP
From: (User #71)
Subj: CLOCK CARD

Do you still need a clock card for the BBS. If you do need one, I have a street electronics Business Card I would like to give away. The

only problem with this card is it phantoms out slots (That's what I remember, I haven't used the card in two years) but it does put out a ProDOS time. Let me know if you can use it.

Nope, we already received a Thunder clock from another subscriber... Thanks anyway... <<<SYSOP>>>

Note #46

Date: 12 Mar 90 (19:11)
To: ALL
From: (User #831)
Subj: HELLO

HELLO I LIKE TO KNOW DOES ANY ONE LIVE IN CALIFORNIA AND I CAN GET YOU ON A FEW GOOD APPLE BBS AND IBM IF YOU AND ARE YOU ON MEDCOM IF YOU ARE I AM PIONEER AND I CAN GET YOU A FEW PEOPLE THAT WILL HELP YOU WITH ANYTHING OK THANKS SEE YA TELL ME IF ANY ONE IS IN CALI AND IS ON MEDCOM OK SEE YEAH

Note #47

Date: 12 Mar 90 (20:08)
To: ALL
From: (User #833)
Subj: HELLO...

I'm calling from San Diego, so you won't see me up here more than once a week, do to expensive phone bills and the like, but I was just wondering if any other San Diego users are on the system yet?? Also, if anyone is on networking sites, you can reach me on a few pro-line boards... My addresses there are:

alank@pro-beagle
alank@pro-grouch

I am also thinking of putting up a bbs down here in the SD area, in which I will support COMPUTIST subscribers and the like.

Alan

P.S. If anyone is thinking of putting up a bbs soon, lemme know because I have written a VERY good bbs system from modemworks, which BLOWS away almost everything I have seen so far in the Shareware/PD scene...

Note #49

Date: 14 Mar 90 (17:09)
To: ALL
From: (User #1116)
Subj: APPLE IIGS LIBRARY!

Hi, 'yall!

I just logged on here for the first time, and, yes, I DO subscribe to Computist. I would appreciate some mail from you guys because I feel a little lonely at the moment.

Anyway, how many of you out there would like to see an Apple IIGS-specific file library? On many other BBS's there's one, and it would be great if there was one... What do you people think?

Thanks for reading!

—Eric Mumford

Note #61 (Reply to #49)

Date: 21 Mar 90 (18:29)
From: (User #1093)
Subj: APPLE IIGS LIBRARY!

Eric, I am relatively new to Computist (since issue #68), and new to BBS's (since the beginning of March). I would LOVE to see an Apple IIGS specific file library - I have not really much to upload, but would upload what little I do have. If you write back, but don't see me respond, it's because I am calling from Connecticut and it is relatively expensive to call here, but I will call back.

Reply: You will in the future (soon?)... <sysop>

Note #64 (Reply to #61)

Date: 22 Mar 90 (17:58)
From: (User #1116)
Subj: APPLE IIGS LIBRARY!

You think phone expenses are bad from Conn... try calling from New York! Horrible... I have an Apple IIGS and, well, I have LOTS of things to upload if David Goforth would only validate them as I uploaded them. I still have to pack them, etc... What local BBSs do YOU call in Conn.? I could probably call THEM and we could communicate. A popular Apple board up here in 'york is Plain Vanilla: 1-518-462-5953. I call there often: it's an excellent board.

I'm gonna check out the file section and get outa here. The phone bill is ringin up higher and higher every minute... See you later!

—Eric

See the TIME SAVER NEWS at the end of this <sysop>

Note #54

Date: 17 Mar 90 (06:30)
To: ALL
From: (User #502)
Subj: CHUCK YEAGER'S AFT

I sent some notes up to the file base on Chuck

Yeager's Advanced Flight Trainer. If there are any questions, send me a message.

Note #53

Date: 16 Mar 90 (23:54)
To: SYSOP
From: (User #576)
Subj: GBBS

My home BBS uses GBBS and it is by far the best Apple software around. We have at least 400 users. I just remembered that I have also seen a very slick new BBS software by a programmer in CA. One of the local boards here in Wichita is testing it (it is Net capable).

Scott

Note #57

Date: 18 Mar 90 (10:33)
To: SYSOP
From: (User #1126)
Subj: SUBSCRIPTION

I am a subscriber to COMPUTIST, but did not receive my ID number in issue #71 as the article described. The number on my mailing label is 0215 if that is of any use. Please advise me as to how I can gain full access to the board as a subscriber.

Thanks in advance
Don

Increased access is granted to all current subscribers (which you now have), FULL access requires that you be a BBS member \$10 annual fee sent to Computist (same address as subscriptions) or calling with Visa/MC to same phone # as subscriptions... <<< SYSOP >>>

Note #62

Date: 21 Mar 90 (18:46)
To: SYSOP
From: (User #1093)
Subj: \$10 ACCESS

David, I sent my \$10 into Computist on a form that they sent to me a week ago. I was curious to see whether you had received the money and updated my access. I would like to know, however I shall not be back until another week has passed, for this is long distance, and I do not like giving that much money to the phone co. that often. Please respond. Thanx, Bret.

Please allow 7 days after mailing for validation <sysop>

Note #67

Date: 23 Mar 90 (20:32)
To: (User #1062)
From: (User #1082)
Subj: CRACKS

Jim,

I'm having a heck of a time with a program called Accelerated Reader.

The author even has a message in the sectors that you'll have to work hard to see this one in COMPUTIST.....would you like a try at it? I've been meaning to call or write to you about it for a long time. Also I have Toy Shop and Bureaucracy that I'd like to see finally cracked. Thanks
Jack

Note #68

Date: 24 Mar 90 (05:27)
To: ALL
From: (User #357)
Subj: WASTELAND CRACK

I have had Wasteland for about 8 months, and LOVE the game. But it requires you to start with the key disk and this week mine started acting hokey!! I desperately need a crack for WASTELAND by electronic arts before I lose it forever.

Thanks
Starbuck (ohio)

Note #69

Date: 24 Mar 90 (18:17)
To: ALL
From: (User #675)
Subj: DARK

ACTUALLY this is aimed towards Ed Teach.. But Does anyone know how to Disable the Rune Check in Dark Heart Of uKKrul?

Also.. look on your local bbs for a 3.5 edition of prince of persia!. a friend of mine transferred the Sides onto a prodos 3.5 disk and added more rooms and a new lvl!

laterz
jim

Feedback (To SysOp)

Note #4

Date: 3 Mar 90 (10:26)
To: ALL
From: (User #726)
Subj: QUESTION

Just a question - will you be providing an INDEX, say in AppleWorks DataBase format for Computist??? That you can download?? I have an older one from an Ad one of the subscribers put in a long time ago, but it is kinda old now and

I just haven't (yeah, I know, lazy) gotten busy to update it myself!!

Joe Oliver

Re: It's available now in the EDITORIAL file base area... <sysop>

Note #5

Date: 3 Mar 90 (16:21)
To: SYSOP
From: (User #452)
Subj: DOCS

I was really pleased to see that I have full access to the Computist BBs right at the start. Calling from WI, I get tons of line noise so you might see some erratic stuff. Anyway, is there some way I can get docs on the BBS software you use? Isn't Appletnet shareware or freeware. If I knew more about the system, I could save some time (read \$\$\$) on the board. Please let me know what you can do and what I can do to support this project

Kindest regards,
TOP GUN

All the software & docs are on the Gopher Hole BBS (206)584-6621 (same cost as calling here) but would take up quite a bit of space on this system since there are several versions (not counting this one)... Up till version 1.3f it was free (2.x is now shareware)... As for saving time use the copy buffer to save all displays to disk as fast as possible & review them at your leisure with a text editor. <sysop>

Note #6

Date: 4 Mar 90 (12:12)
To: SYSOP
From: (User #1079)
Subj: STUFF

Nice to finally see the BBS up. I already subscribe and would like full access. How do I go about this? Also, I'm not too crazy about REAL names since I don't even use one on my own bbs.

Hopefully things will grow around here. You may want to look into GBBS Pro as BBS software, as it runs compiled and a LOT faster. Anet seems like c-net for the commodores. Also look into the U.S. Robotics deal for a 9600 baud modem. Can't beat 'em. Since I've been running a BBS for several years now I'd also like to offer any help that you may need, I just would like to keep the LD bills to a minimum.

I'm getting a copy of GBBS to try out & may put it up (if the cost allows).

As for the 9600 baud (if we can't afford GBBS, it may be awhile). Your real name is only required at Logon to verify you as a Subscriber or BBS member (you can change your handle or nickname in the "P"ersonal data section from the Main Menu. About the best help I can get right now is finding a routine that will allow YMODEM protocols to be added... <<< SYSOP >>>

Note #11 (Reply to #7)

Date: 9 Mar 90 (20:48)
To: SYSOP
From: (User #1079)
Subj: STUFF

WELL, since I have accounts on both GENic and AOL I'll keep you posted if I see anything that may be of help with anet. As for verification, my subscription is under the name Paul Giguere, I have submitted a few articles in the past and have subscribed since about issue 20.

Nice to hear about GBBS!!! I've done a LOT of modding on that system. Feel free to ask, as it is a bit weird till you get used to it.

Know of any patches or extras that would allow support of GBBS with the Datalink 2400? <sysop>

Note #15 (Reply to #14)

Date: 11 Mar 90 (00:15)
To: SYSOP
From: PERFESSER (User #1079)
Subj: STUFF

The people I talked with seem to think that either the CTS 2400 or 2424A driver OR the multi-speed driver should work. What version of GBBS and which ACOS are you planning on using?

Note #9

Date: 8 Mar 90 (20:37)
To: SYSOP
From: (User #1082)
Subj: RDEX SECTION

Is this a section for new deprotects? I'd appreciate access to it if it is. Thanks

This section is for Feedback, leaving the SysOp a message... <sysop>

Note #10

Date: 8 Mar 90 (20:39)
To: SYSOP
From: (User #1082)
Subj: GBBS

GBBS is on its way. I sent it UPS on Monday...the clerk told me it should arrive by Friday...ha ha I'll bet....Anyway good luck. If I can help with anything, even non-BBS related, give me a yell.

Note #17

Date: 18 Mar 90 (00:27)
To: ALL
From: (User #457)
Subj: ADD A MODIFICATION

I think you should add a read text/appleworks WP file function in the file transfer area. This would be easier for some to read short text files rather than download them. Most term programs have some sort of capture feature and it would be faster. Just thought I would pass this along.

Re: If the file is an ASCII text file (shown by a file type of TXT) then you can currently select 'D'ownload and answer 'A'scii to the protocol selection and the file will be displayed just as though a VIEW command had been given. A true view command will be added later (either on ANET or GBBS)...

PLEASE DO NOT SELECT ASCII PROTOCOL FOR SHK FILES - YOU WILL RECEIVE GARBAGE & CANNOT HALT THE PROCEDURE UNTIL THE ENTIRE FILE IS READ!!! <sysop>

Note #18

Date: 21 Mar 90 (16:39)
To: SYSOP
From: (User #1129)
Subj: HELLO...

I am a subscriber and I did not receive the slip of paper mentioned in the article. The Address I left is the one the post office decided that they would give to me when they changed it rather than the box # associated with the current subscription.

Use the Personal data section from the main menu & ensure that your address matches that which you use for Computist. Validation is done by verifying Name & Address (especially zip code) against their data base. Read the BBS Layout bulletin to see what access you actually have (you should be able to get into all areas granted to Subscribers - Increased). <sysop>

Time Saver News

1. Before dialing (or right after logon), set your Copy Buffer ON.

2. Quickly go through all new messages in each message base as fast as possible, DO NOT READ THEM. Do the same in the Files areas, Bulletins, & On-line sections (listing what is there)...

3. You do not have to wait for the menu to be displayed in the Main, Files, or Message areas before making a selection. These areas support 'HOT KEYS' (immediate execution of a command). Also go to the 'P'ersonal data section to view the menu.

4. Then LogOff, save the buffer to disk & review it at your leisure. If you wish to reply to a message, write the message as a text file and save it (be sure to make note of the original message base and message number). Figure out what files you'd like to download from the listings and make note of their Base # and filename. Once this is done then:

5. Call back & get your files. Go to the message base that you want to write to. If a reply, select 'R'ead - 'I'ndividual - #, and then 'R'eplay and send your text file using ASCII transfer. BE SURE THAT YOUR TEXT FILE CONTAINS NO BLANK LINES (include at least a single space before the carriage return <CR> if you need an empty line) as the system uses this as an end of entry marker. Then enter a blank line and select 'S'ave.

6. The Message base also supports a 'F'ile tag system (which must be used if the message is more than 79 lines long). This works as follows: Select 'R'eplay (or 'W'rite - to user # - subject), enter a brief statement (at least one character MUST be entered prior to a <CR>), enter a blank line and select 'F'ile. Enter the filename of the file you wish to attach to this message and send using XMODEM (XModem ProDOS is recommended if available). After the transfer is complete, then select 'S'ave. If using this method, the file may be shrunk (archived) prior to sending for even faster transmission.

Example: 1st call: Screen shows 'Connected xxxx baud at time'.

Press ESC to bypass the intro screen.

Enter user # and password.

Press ESC again to bypass the NEWS.

All new messages addressed to you will be displayed (if private) or indicate the base & message # (if public). Press 'N'ext after each message is fully displayed & the menu appears at the bottom of the screen (we'll read and answer them later).

As soon as the Main Menu starts to appear, press 'MRN' (this will take you to the 'M'essage Base and 'R'ead all 'N'ew messages. When you

see "Last message read", press '4RN' (this skips the E-MAIL and FEEDBACK bases which you read at logon and goes to base #4 and again 'R'eads all 'N'ew messages). Repeat this number sequence until all bases (that you are interested in) have been read. Remember to skip E-MAIL, FEEDBACK, & RDEX (these are private only & you will see anything here for you at logon, the rest ya can't view anyway).

Then press 'QFF'. This will 'Q'uit the message base (returning to the Main Menu and then take you to the 'F'iles area and list all 'F'iles. When the last file is displayed (indicated by "Press RETURN to continue") press RETURN then '2F' (to move to the next file area and list the files there). Repeat this thru all the file bases you wanna see.

If you wish to read the bulletins or go to the On-line section, press 'Q' followed by 'B'ulletins or 'O'n-line. These areas do not support the HOT KEYS (except for 'Q'uit, 'H'ang up, & 'T'ime stats), so you must wait for the menu & the command line to appear before making a selection.

Once you've done this, press 'HY' for 'Hang up'? 'Y'es! You now have all the file listings, new messages, and anything else you want and spent less than 10 minutes on the BBS.

Now save the copy buffer, pull out your favorite word processor and read what you've got. If ya wanna answer any of the messages or leave messages then write your message and save it as a text file - BE SURE THERE ARE NO BLANK LINES (must contain at least a single space before the <CR>) - and save it. If it's a reply, then make note of the original message base & message #. Also make note of any files that you may want to download and the base that they are in.

2nd Call: Call back using the ESC features to bypass the intro stuff. Enter the message base that you'll be writing to (ex: M3) as soon as the Main Menu starts. If replying press 'RI#' for 'R'ead - 'I'ndividual - original message number. After the message is displayed & the lower menu appears press 'R' for Reply. This way you need not know the user # and it also sets up the thread routine (discussed later). If not replying, press 'W#' for 'W'rite to user number and enter the topic or title of the message & verify it as being correct.

Now to send the file. Set your terminal program to send ASCII text & send the file you saved. It will be as though you were a super typist and transfer the file as text (you MUST use ASCII or the copy buffer if you're experienced). The file must be less than 79 lines at 80 (or less) characters per line (for more char's per line, use less lines). After the file is transferred (just a few seconds) select 'S'ave. Repeat this for each message, remembering to change bases as necessary.

Next, press 'QF#' for 'Q'uit the message base - goto 'F'ile area - base number (not necessary for

base #1, also if less than '10' then you must press CR.) Select 'D'ownload and enter the filename you want to download. Select XModem for protocol (ASCII protocol will ONLY work for TXT files. ATTEMPTING TO USE IT WITH ANY OTHER FILETYPE WILL DISPLAY GARBAGE & CANNOT BE STOPPED UNTIL THE ENTIRE FILE IS TRANSFERRED, NO HANDSHAKING IS PERFORMED!!) Set your end up to receive the file (XModem ProDOS is recommended, but Standard and CRC will also work.) Repeat this for all the files you want.

That's it... Now for an overview of all the commands and menus used with this system (it may be awhile before we go to GBBS so hopefully this won't be in vain.) Use the following with the BBS LAYOUT diagram.

Main Menu

This is the artery that takes you to other various areas of the BBS.

Time Stats: Shows your time limit on the system used and time remaining for this call. Works the same in all areas.

Goodbye (Hang up): Logs you off the BBS. Works the same in all areas.

Quit section: Exits the current section and returns you to the Main Menu.

Call the SysOp: Pages "ME" if I'm there, I usually answer if at all possible. Please limit this to necessity and have specific questions (know what you wanna ask). I try to make this as short as possible because I'm usually very busy and trying to be courteous because "YOU" are paying for the call.

?: A question mark in most areas will redisplay the current menu. The main reason for this feature is that you will soon be able to disable the menu displays from the personal data section.

M/F/P/B/O: From the Main Menu you will be taken to the appropriate subsection (Message area, Files area, Personal data, Bulletins, & On-Line programs respectively).

List other BBS's: Lists other bulletin boards with an option to 'A'dd to the list.

Help: Displays what each command (key selection) means & does.

Personal Data: Each user can change his/her handle or nickname (this does not affect your real name displayed at logon), your address (this will NOT change your subscription address, you must notify Computist), screen width (normally 40 or 80), screen height used for pausing the display (recommend zero which will disable it for the fast process described above), phone number (please check this for accuracy), computer, age, specialty knowledge, file enclosures (should be 'Y'es to allow tagging messages with files), carbon copy (allows sending duplicate messages to different users, normally 'N'o as it will ask after each private message if ya wanna send another copy), and the user list (to list all users on the system).

Bulletins: This is where news about the BBS is placed, it is a read only area. To read any message just type the number of the bulletin (and press return if less than two digits).

On-Line: These are external files that allow for various additions to the BBS. Here you can renew or subscribe to Computist or view the on-line index (currently under revision). A message "Program not available" will appear if it is not currently active (this does not mean you're denied access but that it's an upcoming addition - if you see it, you have/will have access). To run any program type the program number (plus <CR> if less than two digits).

Message Base

This is where you read/write all your correspondence, letters, inputs, etc. It is broken down into several sections with descriptive titles.

List: will display all the available message areas. To change to another message area simply enter the number of the base.

Read: will display a submenu for selecting what messages you wish to read.

(New) since your last logon, (Forward) starting at a specific message number, (Reverse) starting at a specific message number, (Selective) messages written to/from a certain individual, (Marked) messages written to you, (Individual) read just one message by number, (Abort) don't read any.

When you read a message it will show a header indicating, Area (with message # and indicate if private or a reply to another message), Date written, To, From (with user #), Subject, and File (if there is a file attached it will also show how many blocks there are), and then the message.

After displaying the message, a menu will appear at the bottom of the screen. This includes: (Again) to redisplay the message, (Reply) sends a reply to this message and threads it, (Delete) this message if to/from you, (Next) reads the next message in forward or reverse order as you selected, (File) sends the attached file shown in the header to you via XModem, (Thread) finds the message that this is a reply to & displays it, (Quit) returns to the Message base Menu.

Write: you must specify the user number you are writing to (0 = All, 937 = SysOp) use 937 for all letters to the Editor, Charles Haight, Computist, etc. & specify who it's to in the 'subject' space. You may press <CR> at this prompt to list the users, this will then ask if you'd like to start the list at a specific user #. Answering 'Y'es will prompt you for the starting user #, 'N'o will give you the opportunity to search by name (or partial name). Pressing <CR> by itself will list all users.

After entering the user #, you'll be prompted for a subject, after that enter your message (75 line max). For blank lines press SPACEBAR <CR>, a <CR> by itself will indicate that you are through entering your message & a menu will

appear at the bottom of the screen. Now you can (List) the message with line numbers, (View) the message without line numbers, (Continue) adding to the message, (Edit) a line in the message, (Replace) specified text in the message with new text, (Save) the message, (Abort) quits to the Message base Menu, or (File) attach a file with this message. Attach files BEFORE saving and remember to save the message AFTER sending the file.

Note: Attached files MUST be sent/received using XModem only!!! Text for the message MUST be sent using ASCII and DO NOT select "F" ile before sending a message in ASCII.

Scan: will ask for the starting message number, then list info about the messages giving, message #, To, From, Subject, and Date. It does not display the actual message.

Delete: will delete a message that is written to or from you.

File Base

This is where all files for downloading are and where you upload files to. It is broken down into several sections with descriptive titles.

File List: Lists all files in the current base available for downloading.

Search: Search the file list for specified data (will find all occurrences of a specified string).

List: Lists available file areas. To change to another file area simply enter the number of the base.

Upload: Send a file (you to me) in either XMODEM or ASCII.

Download: Receive a file (me to you) in either XMODEM or ASCII.

Note: ASCII protocol can ONLY be used with standard (uncompressed) TEXT files. Use XMODEM for ALL other files. This includes both uploading and downloading. AWP and ADB (appleworks) files must be sent using XModem-ProDOS or must be compressed before sending. If an appleworks file is sent in it's original format using any other protocol, it will turn out as garbled text and be unusable.

Examine: Shows contents of a specified file that has been compressed with ShrinkIt.

About the BEST way to save money on long distance charges for uploads is: If you happen to be a member of America Online you can put your uploads there and leave mail to "Go4th" or "AKA Gopher". Or if you have numerous files (or lengthy ones after they're compressed), send them on disk (5.25 or 3.5) to:

Computist BBS
c/o David Goforth
11015 Old Military Rd SW
Tacoma, WA 98498-1524

Please include a short note as to what you'd like me to do with them (you need not be explicit).

Computist BBS — An AppleNET BBS

Main Menu

(M)essage areas & Mail	(B)ulletins and articles
(F)ile section (uploads/downloads)	(O)nline programs
(P)ersonal data & User list	(T)ime statistics
(C)all the SysOp '6pm - midnight'	(G)oodbye (hang up)
(L)ist other BBS's	(?) to display menu
(H)elp with these commands	

Bulletins Menu

(#)Read the bulletin	(L)ist bulletins
(G)oodbye (hang up)	(Q)uit the bulletin section
(T)ime statistics	
1. ABOUT THIS BBS	2. HOW TO DOWNLOAD
3. HOW TO UPLOAD	4. MEMBERSHIP
5. NEWS	6. WELCOME
7. SUBSCRIBE	8. BBS LAYOUT
9. LEGAL RIGHTS	10. CREDITS
11. COMPUTIST	12. RATES

Command: 8

Press ctrl S to stop and start text
Press esc to skip ...

1 = Limited Access, 2 = Increased Access, 3 = Full Access

Message Bases	File Areas	On-Line Programs
1.General (Public)	Public Access	Subscribe On-Line (*)
E-Mail (Private)	IBM & Non-Apple II	
Feedback (Sysop)		
Classified Ads		
RDEX Inputs (Write Only)		

2.Subscribers Only	Editorials	Most Wanted List
	Playing Tips	On-Line Renewal (*)
		Back Issue Order (*)

3.Members Only	Hardware Projects	Computist Index
	A.P.T. (**)	IOB Controllers (*)
	Softkeys/Bit Copy	On-Line Mail (***)

And any other areas/features that may be added

(*) Requires Visa or Mastercard

(**) Advanced Programming Techniques

(***) We will forward your letters by US Postal Service to other

Computist subscribers - Members should read message #2 in the 'Members Only' message base for more info.

Online Programs Menu

(#) Run the program	(L)ist online programs
(G)oodbye (hang up)	(Q)uit to Main Menu
(T)ime statistics	
1. I want to join!	2. Most Wanted List
3. Renewal (Visa/MC)	4. Back Issue Order
5. IOB Controllers	6. Forward Mail
7. Computist.Index	8. File Validation

Personal Data Menu

(H)andle or Sir name	(P)assword change
(L)ocation (City, State)	(A)ge
(W)idth of screen	(S)pecialty knowledge
(T)ype of computer	(#) Phone number
(C)arbon Copy	(F)ile enclosures
(?) display this menu	(@) Time stats
(U)ser List	(*) Page Pause
(G)oodbye (Hang up)	(Q)uit

[Area 1 of 7] General (Public) (89 msgs, 5 new)

Shows current area (1 above) of all areas authorized (7 above) and the title if the base (GENERAL above - with ALL messages being public). Will also show the total number of messages in this base & the number of new messages since you last logged on (this will NOT appear in Private only bases such as E-Mail & FEEDBACK).

Messages Menu

(R)ead messages	(W)rite a new message
(S)can all messages	(D)elete one of your messages
(L)ist message areas	(T)ime statistics
(G)oodbye (hang up)	(Q)uit message section

Command: List bases

Name

1 General (Public)	
2 E-Mail (Private)	← will not get published!!!
3 Feedback (to SYSOP)	← User # 937
4 Classified Ads	
5 RDEX Inputs	← will get published in COMPUTIST!!!
6 Subscribers only	← Both Public & Private messages are allowed
7 BBS members only	← Both Public & Private messages are allowed

Unless otherwise stated, all message bases (except E-Mail) may be published in Computist magazine (Unless the message requests that it not be published - will as a feature for automatic selection)...

[Base 1 of 8] Public Access

Base #'s & titles work the same as the message base.

Files Menu

(F)ile listing	(S)earch the file list
(U)pload (you → me)	(D)ownload (me → you)
(L)ist file areas	(T)ime statistics
(#)Move to Base	(E)xamine file contents
(G)oodbye (hang up)	(Q)uit the file transfer section

Command: List file bases

Name

1 Public Access
2 IBM & non-Apple II's
3 Editorials ← Computist Index HERE!!!
4 Playing Tips
5 Hardware
6 A.P.T.
7 Softkeys & Bit Copy
8 Temp

New files areas will soon be added for: DOS 3.3, PRODOS, & GS ONLY...

Command: Examine Archive (This works only on SHK & BXY files)

Enter the file to examine: amdos

NuFx Archive... (indicates an SHK file & displays the following.)

Date of Creation:	11-Aug-1989 25:25pm				
Date of Modification:	11-Aug-1989 25:25pm				
Name	Type	Blocks	Modified	Compaction	Endfile
New.disk	non	1	<none>	563400%	1

Type "NON" indicates a shrunk DISK (either 5.25 or 3.5) & the 'Compaction' works for all files but not for disks. Will fix this in the future to indicate which size is being used (as a rule of thumb, all GS should be 3.5 and everything else 5.25)...

Command: Goodbye

Hang up? Are you sure? (Yes or No): Yes

Thank you for calling Computist BBS

Time spent: 10 minute(s)

Goodbye ...

You can LOG OFF from any section of the BBS with the "G" oodbye command.

The PRODUCT MONITOR

smooth-running, surprisingly user-friendly interface. (What? You mean all that '5.0 mouse-and-windows stuff works?!) Victory Software's first release IS bound to deliver a hefty dose of encouragement to many would-be IIGs game designers. Meanwhile, players can 'sign up' for 30-40 hours of solid adventuring and a good story in "2088: The Cryllan Mission".

- RATINGS**
- ☆☆☆☆☆ SUPERB
 - ☆☆☆☆ EXCELLENT
 - ☆☆☆ VERY GOOD
 - ☆☆ GOOD
 - ☆ FAIR
 - ☹ POOR
 - ☹☹ BAD
 - ☹☹☹ DEFECTIVE

Mushrooming Software

As observed last issue, numbers and quality of II series entertainment wares has never been better; and, the same holds for utilities, productivity products, and educational packages. Incredibly, after more than three years of less than enthusiastic Big Green attentions, the IIGs is in real 'danger' of emerging, overall, the best-supported of any super-res "personal computer". This month's column, the biggest PM ever, samples just a bit from each category.

2088: The Cryllan Mission

\$69.95 Victory Software

☆☆☆

Requirements:
1.25 MB Apple IIGs,
one 3.5" drive.
Second drive recommended.

In the summer of 2087, Earth's fledgling Space Agency began to receive a series of startling reports from an exploration craft in the Tau Ceti sector. Not only had the crew of the S.S. Houston discovered an Earth-type planet, "Crylla", inhabited by peaceful, friendly humanoids, they had actually deciphered the language and been assigned an ambassador. By the time contact with the explorers had to be suspended (Crylla moved 'behind' its sun vis-a-vis Earth), Space Agency scientists could hardly wait for the knowledge windfall sure to come. But they did wait... and wait... and wait. Needless to say, the follow-up "Crylla Mission" which makes planetfall barely a year later is not the delegation of expectant researchers originally planned. With weapons specialists, science officers, and a doctor, your party of six trouble shooters is grimly determined to find out what happened, and, if necessary, shoot the trouble!

When, some centuries later, Star Fleet draws up the guidelines which become known as the "Prime Directive", you can bet that someone will bring up Crylla to drive home the hazards of cultural contamination. Just why is one of your team's discoveries as you piece together the remarkably verbose stories, hints, and social commentaries of Cryllan citizens, surviving Terrans, and others you encounter. Yet, with a few easily sensed exceptions, what you discover in conversations has less real impact upon ultimate success than upon 'atmosphere', level of involvement, and overall enjoyment. "Cryllan Mission" 's extensive prose adds a thought-provoking depth unique in an Ultima-type map/maze scenario. (Hint: 'Talk' to everyone, but avoid copious note-taking.) As to how Earth's ace 'fact-finders' deal with what turns out to be a cultural disaster: that is something you determine in a quest spanning the cities, ruins, and labyrinths of two continents.

Crisp top-down countryside maps, detailed cityscapes, and self-mapping 3-D perspective forward-view mazes give "Cryllan Mission" a 'look' and feel very reminiscent of SSI's popular "Queston II"/IIGs. This time, however, you guide the development of six multi-attribute characters, deal with elaborate weapons and armor systems, and direct your party in challenging, partially animated, tactical combats against a variety of opponents. Unfortunately, while the latter can be very entertaining, an overly generous character development scheme results in near invulnerability and ample (even excessive) wealth about midway in the adventure. Opting for "computer controlled" combat helps move things along; so, you are able to concentrate upon exploration and the location of items needed for mission completion.

Featuring speedy GSOS 5.0 windows for maps, text, and monster/mazescape displays "Cryllan Mission" employs virtually an 'out of the box' format, and, nevertheless, achieves a

Postcards

\$29.95 Activision/Mediagenic

(Additional blank cards: \$7.00/100)

☆☆

Requirements:
Apple IIGs,
one 3.5" drive,
Imagewriter or compatible printer,
super-res paint utility.
Color ribbon recommended.
(64K II version available)

Summer, the #1 postcard-sending season, is just around the bend; and Activision's "Postcards" could be the perfect medium for showing off your artistry, your wittiness, and your computer, all in one fell swoop. Available in separate versions for IIGs (color super-res) and II+/IIE (\$24.95, B/W hires), the package is, blatantly, something of a gimmick. Included on-diskette are ten unpopulated scenes (beach, desert, moon-scape, ...), plus more clip-able artwork for Structures (Taj Mahal, Eiffel Tower, ...), Objects (umbrellas, Easter Island Stoneface, ...), Transportation (blimp, flying saucer, ...), People (hula dancer, alien, tourist, ...), and flocks of Animals—about 150 pieces in all. The IIE package adds a simple cut & paste drawing/printout utility; IIGs users are expected to rely upon a full-featured painter, like Activision's "Paintworks Gold".

Admittedly, the artwork IS cute and distinctively 'post-cardsy'; AND—the gimmicky part—you also get 25 blank-front 6.5" x 4.5" postcards with (on back) "Place Stamp Here" box and "Postcards by Activision" along the card center. PLUS, there's a glue stick for pasting your printouts to the card fronts! (Clever; however, if Activision supplied a tractor-feed card carrier sheet, the glue wouldn't be necessary and post-card production would be easier.) Assuming you don't re-size the on-screen display, a finished product (e.g. after pasting and trimming) measures 6" x 4".

My first card—the standard waving space creature on a city street with "Wish You Were Here!" caption—turned out so well I almost hate to send it. Supplied with idea-packed manual, clip art listing, and order blank for more cards, "Postcards" makes bragging about your vacation more fun than ever.

Geometry

\$79.95 Broderbund

☆☆☆

Requirements:
512K Apple IIGs,
one 3.5" drive.
Second drive recommended.

Intended for textbook support/enrichment, Broderbund's new educational package takes the user from elementary concepts, through Plane and Solid geometry, to the beginnings of analytic geometry. "Points, Lines, Planes", "Congruence", "Right Triangles", "Areas and Volumes of Solids", and "Coordinate Geometry" are representative of the ten chapters which, not counting separate problem and proof displays, average some 25 super-res screens apiece. To better individualize each presentation, users may select just a unit's "Tutorials" or "Problems", or both. In addition, a "Classical"/"Simplified" setting adjusts material to more closely follow approaches used in currently popular textbooks.

Rendered in dithered-color 640-mode, "Geometry" delivers attractive textbook-look screens with boldface and color highlighting to enhance legibility. A simple click on a corner 'turns' the page (forward or back) at, roughly, 5-10 seconds per update. Not, to be sure, 'book speed'; but, then, you can also hop quickly to any part of the presentation—both "Contents" and a comprehensive "Index" are always on-line—and, the 'computer pages' ARE a good deal more active than pulp and ink. Aside from pull-down "Reference", "Hint", and "Solution" options, you will often find click-able text in green highlighting to flash the lines of an "Acute Angle", fill-in "Base Area" of a figure, etc. Other buttons may show a proof or call up "More" discussion.

No textbook, of course, can offer "Geometry" 's user-changeable figures (that is, unless you've been staring at a proof too long). This powerful learn-by-doing aid lets you mouse-diddle line relationships (e.g. in triangles, a rotating quadrilateral, etc.), 'grow' and stretch solids, and much more, all right on the 'page', while obtaining continuous numeric/descriptive readout(s) of key figure parms and characteristics!

As you may recall from your last encounter

with math studies, there is no such thing as an error-free math text. True to this tradition, "Geometry" 's developers occasionally allow a single drawing to stand for three or four problems, with the result that angles shown as obtuse are actually acute, lines which must be parallel are shown diverging, etc. I've seen just one clear case of mislabeling; but, in the "Area of a Trapezoid" discussion, one finds $.5 \times h \times (b1 + b2) = .5 \times (h \times b1) + (h \times b2)$! Mildly disconcerting; still, considering quantity of material presented, the few 'obvious' errors fall within normal "for the student to discover" limits.

Fine as a 'quickie' reference and stand-alone refresher (e.g. for SAT prep), "Geometry" shows to best advantage as a 'light up your lines', interest-stimulating support resource. Including "QuickPrint!", a handy super-res screen dump CDA, the complete course comes on three mini-diskettes with handsome 25-page User's Guide.

Gnarly Golf

\$29.95 Britannica

☆☆

Requirements:
512K Apple IIGs,
one 3.5" drive.

The last time I checked, our trusty IIGs had failed to sprout a coin slot. So, why does Britannica's new 18-hole mini-golf arcade (for 1-4 persons) eject players who "run out of strokes"? If, as seems likely, the aim is to encourage replays, then "Gnarly" designers have miscalculated. Like the "Zany" 'guys at EA, they've over-valued the lure of 'getting to the next hole' and under-valued just about everything else. As a lone player, I would much prefer to challenge a 'Best Score' and see all the holes. When two, three, or four computer golfers start a game, it's because they want to PLAY—not watch, while the player who owns the computer (and gets to practice) demonstrates his/her expertise.

"Gnarly Golf" does start you with six strokes, award a number of strokes equal to par for each hole finished; and, if an experienced player is first in the shooting order, new players can learn much about how to approach the hole. Aiming is simple—you 'spot' your target with a mouse pointer—and over-powering a hole shot won't send it over the cup. Further reducing the advantage of practice, "Gnarly Golf" 's three-click 'shot meter' (to control power and spin) runs somewhat slower than the ones you'll find in Accolade's games or EA's "World Tour Golf". Remaining strokes and current scores (based upon holes finished, length of holed shots, etc.) are continually updated at the corners of the display; but (another mistake), the game fails to maintain a High Scores roster on-disk.

With the exception of small spot-like balls which show no rolling motion, "Gnarly" offers very good 'production value', including classy 3-D lettering, background music, good physics, a set-your-own-golf-ball-color option, an animated golf ball guy who performs a different routine to set the theme for each hole; and a bizarre set of nicely illustrated single-screen challenges built around pool tables, rat mazes, prison cells, ant hills, toys, and much more. The shot, rebound, etc. sound effects work well enough; but, considering the variety of opportunities for something more elaborate (e.g. water traps, air vents, hot griddles, leaky hoses, flipping trowels, conveyor belts, ...), they qualify as 'just adequate'. Supplied on two mini-diskettes, "Gnarly Golf" is too easy, low-competition stuff for adults, but a cute, entertaining 'starter challenge' for younger players.

Warlock

\$39.95 Three-Sixty Pacific

☆☆☆

Requirements:
512K Apple IIGs,
one 3.5" drive.
Joystick recommended.

Spanning twenty split-level screens (or "Levels"), Three-Sixty's "Warlock" is a super-res arcade adventure showcasing gem-like figures, superb animation, and full-range sound. As the dauntless Warlock, your objective is to collect eight key objects and then wrest the stolen Karma Jewel from the evil HE's clutches.

Eight-direction KB/joystick movement and your staff's fire button are all you need to climb stairways, swing across deadly pools, ... and blast hordes of zombies, ghosts, goblins, etc. (i.e. the 'cannon fodder' class monsters) as well as major league horrors like giant spiders and fire-belching golems. Shots cost Power points and each hit from a monster or trap reduces Vitality (0000=Dead). However, you can start with up to 4000 Power and 5000 Vitality; so, even for the unskilled beginner, there's plenty of play; and plenty more, once you learn to recognize snares and become adept at collecting the numerous Power/Vitality/Armor-boosting goodies.

Forests, graveyards, ruins, underground tun-

nels, lakes, shrines, ...: it would be fun to just 'page through' the colorful, richly detailed gamescape; but you can't. You must EARN your way; and, with no Game Save (an unfortunate deficit), getting beyond Level 7 represents a fairly stiff challenge. Durable fun, "Warlock" is a great super-res/super-sound showoff piece (and, if you need another excuse, perfect for entertaining younger visitors).

Cartooners

\$59.95 Electronic Arts

☆☆☆

Requirements:
1 MB Apple IIGs,
one 3.5" drive.
Second drive or hard disk recommended.

"Cartooners" makes it easy for even young computer users (e.g. ages eight and above) to produce 'for real' super-res cartoons complete with colorful scenery, animated characters, "speech balloon" dialogue, and music and/or sound effects. At up to 2000 frames each, a "movie" can, depending upon user "Speed" setting, easily run several minutes, which means a finished product can pack plenty of entertainment value.

Aimed squarely at the 'computers for kids' market, Electronic Arts' two-diskette package includes both a well-organized, 'serious' 60-page manual, and a cute, easy-reading "Look and Play" booklet. The latter introduces early readers to the program's facilities for movie playback and beginner editing. In "Look" you just click on the ticket(s) of up to six selections to watch. "Play" works much the same; but, you also get VCR-style controls (FF, Rewind, Play, Record) and have access to full editing capabilities (including standard "File" pull-down options for loading any non-ticket movie). Originally, selections are limited to the short-run movies supplied (e.g. "Animal Boogie", "Bad Wolf", "Park Play", ...); but, since you can change ticket names, any twelve movies can be available for easy click-on playback.

The 'trick'—the reason it's so easy to produce cartoons—is that "Cartooners" supplies a large cast of expertly drawn animated actors. All the user has to do is add Mouse, Bunny, Duck, Piggy, Dog, etc. and 'tell' him/her what to do. For instance, you might click on Mouse, place him, and then 'tell' him (via pull-down "Action" choices) to Walk Slowly Right. Now, as each "Record" click advances the frame count, Mouse will perform some walking motion and move to the right. Mouse can also turn one way or the other and face the front. Duck can walk and fly, Bunny can hop, Butterfly flaps, ...; repertoires vary, but, in general, you can select an action (or "freeze") and set speed and direction for each actor whenever you wish. To handle situations where Mouse must walk in front of Bunny, behind a tree, etc., each actor's "Foreground Level" is settable in eight levels and, as with placement and Action, may be changed in any frame.

Lamp posts, bushes, kettles, campfires, and other props are also considered "actors". With few exceptions (e.g. Flower can change color), props can be 'told' to stand still or move around; but, only a very few, like Explosion, are animated. Another kind of "actor", the stretchable speech balloon, comes in "standard", "exclamation", or "invisible" versions, and can include text in various fonts, sizes, styles, and colors.

Obviously, by the time you add one or more "Scenery" backgrounds and several "Music/Sound Effects" pieces—both of which are settable on a frame-by-frame basis—PLUS multi-frame 'Cut and Paste' editing ...; well, a movie CAN become a fairly sophisticated, "slick" production. Music/sound stuff can be imported from EA's "Instant Music", any paint utility can produce backgrounds, and new prop-type actors are available from "Deluxe Paint II" as "Brushes". (There are no utilities to permit user development of fully-animated actors.) Whereas RAM space seems to be the chief factor restricting number of backgrounds and sound selections employed in a movie; speed considerations, ultimately, limit number and complexity of actors on-screen at any given time.

As might be guessed from the 1MB RAM requirement and separation of resources onto System/Program and Art diskettes, "Cartooners" does not function comfortably on 1.25 MB systems having only one drive. With two drives you can easily achieve swapless boot-and-Look/Play operation; but creation of larger movies and scenery/actor/sound additions are bound to entail some planning to maintain developer convenience (e.g. grouping movies with the resources they employ on other "ART" diskettes). Add the fact that movie files are optimized for simplicity, NOT compactness, and 'the cat is out of the bag'. For all its claims of single-drive useability, "Cartooners" merely tolerates two-drive setups and really prefers to run on hard disk.

Boasting a range of creative possibilities far beyond 'kiddie level', "Cartooners" can deliver advanced "knock your socks off" results, yet adroitly manages to make even beginner products 'look good'. From developing pizazz lead-

ins for videotaped presentations—the manual describes the simple IIGs-to-VCR connection required—to entertaining young users, “Cartooners” is consistently fun, consistently classy.

Copy II Plus, 9.0

\$39.95 Central Point Software

☆☆☆

(Current Owner Updates: \$18.00/\$23.00 for manual plus one/both diskette formats)

Requirements:

128K/80-col. Apple II series, one 5.25" or 3.5" drive. Second drive recommended.

Probably, almost no one was surprised when “Copy II Plus” garnered over 80% of the 1st-place “Utility” category votes in our recent “Best New Stuff” survey. Over the years, as Apple introduced new DOS’s, “Copy II” has consistently out-performed Apple’s own disk utilities, served as a reliable bridge among operating systems, and permitted speedy backup of many protected products.

The Version 9.0 package includes UTILITIES and BITCOPY software on both 3.5" and 5.25" diskettes. BITCOPY utilities for both formats and Auto Copy parms for all supported products are supplied on the mini-diskette; the 5.25" diskette includes only the 5.25" BITCOPY and parms for products released in 5.25" format. Using “Copy II”’s new Compare Files function, I was able to quickly verify that “UTILITIES” is identical on the two formats. (Compare Files will check any two ProDOS files and allows you to specify whether or not differences should be displayed or just the number of mismatches.) Other improvements include use of “all available memory” to speed copying, an “any way you like” catalog sorter, and function selection (in UTILITIES and BITCOPY) via arrow keys, letter keys, or mouse.

(BUG NOTE: Regarding the failure of Sort Catalog to update pointers, as reported by Dave Goforth in the last issue... First, Dave deserves a pat on the nose—pat, pat—for zeroing-in on a potentially nasty problem and getting the word out. Central Point is aware of the problem, expects to have it fixed soon, and will supply free replacements to Version 9.0 purchasers. I’ll let you know when mine arrives.)

Naturally, you can count upon finding additions to Auto Copy’s already extensive collection of entries. “Copy II”, evidently, does not drop parms, even for rather ancient products. If an Auto Copy entry produces a “Not Found” message, scroll through the supported products list before deciding a “golden oldie” is missing. (One source of confusion is that, from revision to revision, a product’s name may undergo minor changes; “Bard’s Tale” may become “Bards Tale”, etc..)

As in earlier versions, “Copy II Plus, 9.0” starts up in its ‘standard’ ProDOS and DOS 3.3 UTILITIES mode. Besides Copy and Compare Files, UTILITIES lets the user obtain Catalogs (with or without file lengths, hidden characters, deleted files), Verify the readability of file(s) or diskettes, check 5.25" drive speed, View TEXT (with or without hex values) in a file, Delete files, Undelete files which have not been over-written, Lock/Unlock files, Rename files or volumes, Change the name of a DOS 3.3 boot program, Create sub-directories, and display a file-by-file Map of sector usage. (The latter, however, merely puts *’s on an un-numbered dot grid.) One powerful new capability, being able to launch ProDOS Applications from a user-created menu (saved on-diskette), lets you, in effect, expand UTILITIES functions.

UTILITIES will Format diskettes in DOS 3.3 (5.25" only) or ProDOS (with option for GSOS’s preferred 2:1 interleave). Selecting “Options” allows setting Date (on older machines lacking a built-in clock/calendar), mouse speed, and Printer slot. (Setting the latter automatically causes UTILITIES to offer a “Printout Y/N” choice for Catalog, Compare, View, etc. functions.) All in all, it’s small wonder that “Copy II” always wins space on a hard disk or that II users lacking a hard disk routinely copy UTILITIES.SYSTEM to “System” and “ProDOS Utilities” diskettes. As for handling “protected” software, “Copy II” offers a comprehensive set of editing and copy functions called BITCOPY.

On the Universal Frustration Scale, few experiences rate above having an un-archived diskette bomb in the midst of some design task or at a crucial moment in game play. Anti-copy “protection”, while of dubious benefit in preventing the unauthorized unfair mass distribution of a product, definitely makes archiving more difficult for legitimate purchasers. (An ironic twist is that many protection schemes significantly increase the likelihood of diskette failure and, thus, make archiving all the more necessary.) One possible solution—the ‘standard’ first try—is to run the appropriate BITCOPY (3.5" or 5.25" from UTILITIES), pick Auto Copy, and type-in the target program’s name. If the product is listed—sometimes there will be several “version” choices—and, if the protection hasn’t been changed, there is a fair chance that, in just a minute or so, you’ll have the desired backup.

‘Easy’ alternatives for unlisted products include a series of Auto Copy “Try ...” entries shown alphabetically along with product names. While most are simply convenient Manual Bit Copy presets (e.g. “Try Nibble Count”, etc.) a few relate to specific manufacturers. Sometimes, just a slight modification of an existing parm, via Partial Auto Copy, is the answer. In fact, a check of Computist’s own pages may yield the needed PAC parms or even an entirely new parm set you can key-in to Manual Bit Copy. (Either way, you can use Edit Parm Entry to produce a new record for future Auto Copy access.) Sometimes the easiest approach is ‘none of the above’; and you’ll find yourself cursoring-down to BITCOPY’s Sector Editor.

Once a part of UTILITIES, the Sector Editor was moved to each BITCOPY several revisions ago in order to take advantage of modifiable parms. In 3.5" BITCOPY, selecting the option opens an 80-column TEXT screen showing about half of a sector (17 rows, 16 bytes-plus-ASCII text per row) plus the 12 “tag bytes” used in Mac formats. (Scrolling permits viewing all 512 data bytes.) Pressing “M” toggles between “Block” and “Track/Sector/Side” access. In 5.25" BITCOPY, Sector Editor offers only a 40-column TEXT screen and Track/Sector access to diskette data. Since Sector Editor displays program code and data the way it looks when read-into/written-from memory, you will usually be able to view program messages, names of accessed files, names of adventure characters, etc.. You can peruse diskette contents, obtain screen or disassembly printer Dumps, do editing and write the changes, and Scan the diskette for strings of text or hex bytes.

All of the above, naturally, work fine on “open”, unprotected software. To work with diskettes protected via changed sector address/data prefixes, checksum mods, etc., a Patch option lets you ‘tune’ Sector Editor’s Read/Write routines. Pre-set choices include ‘standard’ and ‘loose-error-checking’ versions of both 16-sector DOS 3.3/ProDOS and 13-sector DOS 3.2, plus “Custom”. As illustrated in last month’s column, with the correct Custom Patch entries in hand, copy protection need not be a barrier to editing adventure game characters. Other applications include elimination of unwanted routines (e.g. squashing a CALL which turns off Control Panel access), direct modification of directories, salvaging of data from damaged files, ..., AND defeating copy protection. Especially on 3.5" media, Sector Editing a few bytes may be sufficient to defuse protection, leaving you with a fully functional backup.

Note: Due to a curious oversight—I called Central Point to make sure—Custom parms for Sector Editor patching must be user-entered each session, whether or not the correct parms are included in an Auto Copy file. (If they are, YOU have to jot down the needed parms and type them in.) Selecting “Edit Parm Entry”, you are free to change Auto Copy parms and create new entries. You can even ‘tell’ Auto Copy to do sector editing as part of its copy process; BUT, as of Version 9.0, you cannot simply save and, later, reload a Custom Patch for use by Sector Editor.

When the protection gets really tough, the Really Tough get Manual Bit Copy. With nearly ninety parameters to manipulate, selecting MBC puts you at the helm of a nuts-and-bolts ‘tailor-to-fit’ copy machine. Supporting your efforts are Hi-Res Disk Scan (to quickly locate tracks and half-tracks with useful data), Nibble Editor (to display/analyze/modify the ‘raw’ bytes actually stored on a track), AND the newly rewritten Version 9.0 manual. Much more than a function-by-function user’s guide, this 189-page ‘bible of backup’ describes parameters, discusses disk hardware and ProDOS/DOS 3.3 formats (including track and sector organization, encoding, interleaves, etc.), surveys protection methods, and adds helpful appendices. Appendix E, for example, is a table for converting Decimal/Hex/Binary to 4-and-4 disk encode bytes and 6-and-2 translate values. Yet another BITCOPY function, Manual Sector Copy is an MBC-like option (with its own parameters) for tackling sector-oriented protection methods.

As indicated earlier, “Copy II Plus” has, for some time, managed to satisfy Apple II users amidst changing operating systems, machines, and media formats. In Version 9.0, however, the ‘one package for all’ idea has very nearly reached its culmination. A 9.1 release might offer a Sector Editor with directly loadable custom parms, an option to copy 5.25" diskettes to IIGs RAM disk, a Purge Catalog option, and a better Disk Mapper. DOS 3.3 users would, most certainly, appreciate a mod to allow manipulating ‘3.3 files without having file names shortened, filled with dots, and otherwise ProDOSed. (If you still own an ancient ‘3.3-only’ “Copy II”, don’t even think about overwriting it.) There IS room for improvement within an ‘all II’s’ approach—well, “all II’s” EXCEPT 64K machines—but, with the advent of GSOS 5.0, not much enthusiasm for tying IIGs “Copy II” performance to IIe/IIc limitations.

For starters, in GSOS 5.0 (\$50.00 for package with manuals), Apple’s ‘stock’ Finder has achieved a marked speed advantage in virtually every copying task:

Copy Disk W/O format:

OS 5.0 = 1:23 vs C2 9.0 = 2:32

Copy a test folder:
OS 5.0 = :08.1 sec. vs. C2 9.0 = 18.6 sec..

(Note: In the above, source and destination media were 2:1 interleave formatted. GSOS was set for 256K of cache. “Copy II” ran with DSR’s Diversi Cache set to 256K.)

Speed, plus super-res displays, windows, and the benefits of being able to work with 16-bit applications using a 16-bit operating system, ... all lead to an obvious conclusion. Version 9.1, 9.2, ..., whatever; Central Point must, eventually, deliver “Copy II Plus/gs”.

Though hard-pressed by competition in some areas, “Copy II Plus” remains a unique, powerful combination of disk utilities—the ‘indispensable tool’ for all Apple II users. Current owners can hardly lose on the update; the Compare Files function, alone, should be ample incentive. As for new purchasers, just a few minutes with the 9.0 manual will erase any doubts. You can wait for the next revision—there will always be a ‘next revision’—but, today, you won’t find a better utilities value than “Copy II Plus, 9.0”.

Fast Frames, Updates, Etc.

Great Western Shootout ☆☆

Back in the ancient days of personal computing, you may have typed-in a neat little SoftSide game named “Blast Away”. In Britannica’s modern version (\$29.95, for 512K IIGs), you get the same point-and-shoot challenge, but with ten super-res Western backdrops, authentic gun battle sound effects, smooth-action mouse-aim crosshairs, second player (alternate turns) option, and targets that shoot back! As in law enforcement ‘training villages’, your objective in each single-screen shootout is to nail animated ‘pop-out’ figures of outlaws while leaving any innocent bystanders unscathed. Slowness and/or misses let the outlaws shoot back and costs Health points, as does hitting a bystander. Zero Health ends the game. (To even the odds, the program occasionally displays a small Red Cross patch. Hitting it restores Health.) Each shootout is good for a minute or so of moderately easy through fairly rough action, with scoring based upon outlaws potted and level. The top four scores are maintained on diskette. Including directions for moving to hard disk, “Shootout” offers quick restart and plenty of addictive action.

Crusin’ on Crylla: 2088

A good guide to party composition is your standard U.S.S. Enterprise “away team”. One doctor can handle all healing chores; a science officer supplies large-area map views (via “Terrain” and “Life Form” scans); and, you need the fire-power of your four weapons specialists. Explore the Ruins thoroughly; the Caverns are good for loot and honing combat technique; take care of things in Adion before entering Cramur. Transports are expensive and you cannot take them into mazes; Heavy Armor is a better investment. The best weapons are won from elite enemy guards. Finally, two items which should have been included in your mission briefing (i.e. the manual): 1. To enter a transport you’ve purchased, the first step is to move onto the transport figure and ‘click’; 2. Any resurrected team member will have reduced attributes. Don’t waste time trying to cure this condition (you can’t); after a day or so the attributes will return to normal values.

Laser Force ☺

This super-res Britannica release (\$29.95, for 1.25 MB IIGs) delivers 75 single-screen tests of your ability to run around, get stuff, and blast robotoid monsters. As in “Spindizzy” and similar arcades, each 3-D perspective gamescape is composed of blocks, ladders, and platforms, with movement permitted in four directions plus up/down. Featuring attractive graphics, decent sound, AND the opportunity to design your own gamescapes, “Laser Force”, nevertheless, manages to bomb in the area most critical to arcade enjoyment: player control. Using a joystick, you discover that an upper-right push equals Forward (i.e. away from you on the game surface) and lower left is Backward (toward you); L/R stick pushes are for L/R movement; and forward/back pushes are for Up/Down. Such a silly, unnatural feeling arrangement is bad enough (especially since, with a spare button to activate Up/Down, the mess could easily have been avoided); the cruncher is that, often, the program simply fails to recognize stick moves. (You’ve got your Flight-Stick—properly centered as per game readouts—pushed all the way to the right; and you just sit there!) Keyboard input is more responsive; but, once again, you must adjust to a wacky layout (“O” is forward, “N” is reverse, ...)—and, in any case, KB is NOT a sensible control choice for this kind of game. No joystick, no deal.

The Legend Of Blacksilver ☆☆

From the creators of “Questron” and “Legacy of the Ancients”, this Epyx single-character maze adventure (\$49.95 for 128K Apple II) sends you into the lands of Bantross to thwart the evil ambitions of Baron Taragas. (Having discovered a vein of magic-enhancing Blacksilver, the Baron

has reinvigorated the monster minions of legendary wizards, added some of his own, and threatens a new reign of terror, chaos, and cataclysm.) Rendered in double-hires, the game sprinkles its large continent with Questron-type cities (where you find shops, gaming parlors, banks, opportunities for profitable mini-quests, etc.) and Legacy-type Temples (where you find the Archives with their magical “exhibits”).

Most of the action is set in five large top-down view castle/labyrinth areas and forty 15x15 3-D perspective forward-view mazes. Advancing your character’s rank boosts attributes, improves skill in using the modest nine-spell magic system, and helps you to survive the numerous non-tactical combats. Mainly, however, the challenge boils down to maze-busting as you collect a plethora of gems, keys and other artifacts, solve assorted puzzles, and contact key personages. (Note: For players who prefer to avoid mapping, Epyx’s “Masters Collection Hint Book” is a bargain at \$7.99.) Two-voice ‘old Apple’ sound/music, double-hires artwork, and a joystick control option are pluses; the somewhat spotty text isn’t. Supplied on four diskette sides, with map and illustrated manual, “Blacksilver” is long-play adventuring fun concluding with a promise: “Your valor may one day be tested again!”

Workers Wanted ☆

Granted, when signing-on for a quest, you anticipate that Evil will do its worst. Hardships are to be expected; and, besides, ‘blood, sweat, and tears’ make for great story telling at the Adventurers’ Inn. Sometimes, however, a very promising challenge bogs down, Evil is upstaged by game mechanics, and you may find it difficult to separate questing from a stunt on the chain gang.

In the tradition of “Impossible Mission II”, “Conan”, and similar ‘extended arcades’, Three Sixty’s “Dark Castle” (\$44.95 for 768K IIGs) challenges you to overcome fourteen obstacle-packed chambers and overthrow the Black Knight. Aside from dealing with the usual rats, bats, spiders, (zombies, flying monkeys, dragons, ...), you will be required to climb ropes, swing across chasms, leap nimbly amongst floating isles, AND collect a few key artifacts and weapons. You’ve got just five lives; but potions help ward off the bites of pissant baddies; and, the place is littered with nice throwing-size rocks, which turn out to be your stock monster poppers.

It can be fun to tackle DC’s super-res ‘puzzle chambers’; but, several flaws tend to put a damper on player enjoyment. Typical of such games, you ‘learn by dying’; and, with no Game Save, boringly repetitious restarts are unavoidable. More bothersome, the program seems to have trouble deciding just where your action figure is vis-a-vis monsters, the background, and collectable objects. Naturally, control suffers; and basic maneuvers, like moving down a ramp or picking up items, are sometimes ridiculously difficult. Good animation, fair sound, and crisp artwork go just so far; expect to WORK for your fun in “Dark Castle”.

Featuring digitized sound effects and four-quadrant movement in a partially-3-D gamescape of well over 100 partially-animated super-res frames, Activision’s “The Last Ninja” (\$39.95 for 512K IIGs) aims to put you IN your very own martial arts movie. Oddly, despite a considerable investment in sound effects and artwork, the game falls short when it comes to matching user controls to action figure movements and outcomes. Crossing any of the numerous stepping stone hazards involves several time-consuming Game Save/Restore sequences, simply because learning the few moves that work costs precious lives and/or wastes game time (thus ‘using up’ Ninja Magic you may need on the other side). One-on-one combats—you tackle assorted evil ninjas, guards, monsters, etc.—demand only a sane level of precision; but, the mundane task of collecting an object in plain view can become a vexing chore as you hunt for just the right position. With puzzles and other adventuring tests to tweak your interest, “The Last Ninja” can be entertaining... IF you don’t mind plenty of hard work.

Bad Duds ☹

Maybe there really is a machine and format that produces a display remotely similar to the one Data East’s “Bad Dudes” promises. At \$34.95, 128K Apple II double-hires is NOT it. Beware of spotty figures, controls with all the precision of a broom handle, and swarms of duds.

Correction: Bums Don’t Star

A mis-print in issue #71 awarded a 1-‘Star’ rating (“Fair”) to “Fast Break”, Accolade’s unfortunate basketball simulation. The rating should have been “Poor”.

Matching Madness ☆☆

Remember all those “find the two that match” workbook you aced-out in grammar school? So does Baudville; and in “Mad Match” (\$39.95 for 512K IIGs) both adults and beginning computists can tackle a super-res/super-sound workbook just packed with ‘click the two that match’ challenges. Each “Level” fills the screen with a series of six-pane problems—ten at Level 1, twelve at Level 2, ..., etc.—shows a bar graph countdown timer, and keeps track of performance for one or

two players. (Two players can share the mouse and take turns or go head-to-head in a 'simultaneous' setup with one player using the keypad.) Other setup options let the user change Time allowed per level and starting Difficulty. "Eagle Eye" leads off with objects like vases and pumpkins; whereas a "Beginner" first encounters simple shape/color matches and works through toys and clock faces.

Success is rewarded via on-screen comments (like "Good", "Awesome", etc.) and leads, generally, to more complex shapes, scenes, and displays (e.g. faces, boxes of crayolas, and music scores). I say "generally" because, especially beyond the lower few levels, significantly 'harder' matching tests occasionally come before easier ones. For best learning, the progression should be steady and fairly gradual. (Better yet, the sequence should be changeable by the user/teacher, with an option to import user-created artwork.) A second weakness is the absence of on-diskette 'player' record-keeping. Still, there's not much question that, for most learners, the package can be an effective aid to developing the Visual Memory and Visual Discrimination skills commonly associated with both reading and math. Add music backgrounds (including a "Break Period" option with 'Optimum Aviary' sound effects) and "Mad Match" is, as claimed, almost too much fun to be educational.

Skate or Die ☆

Boasting two smooth-scrolling, multi-screen raceways, this Electronic Arts skateboard challenge (\$39.95 for 512K IIgs) rounds out the program with High Jump, Stunt, and Pool Joust competitions. Except for the Jump and Stunt half-pipe contests, events allow simultaneous two-player action, one of whom may be human or computer. (Only one human player gets to use joystick control; the other is stuck with keyboard input.) Starting at Rodney's Skate Shop, players can sign-up for rated competition, view High Scores—the top three for each event are maintained on-disk—or just say hello and skate off to practice. The top-down view courses for Downhill Race (through a park) and Jam (through a series of alleys) offer detailed artwork; but, as in the other events, you must endure mushy controls, low-impact sound, and a relentless sameness in action. Too good for "bummer" and too weak for "aggro", the bottom line is Skate or Die or Try something else.

Uninvited ☆☆☆

Running off a deserted country road and smashing into a tree doesn't exactly constitute an invitation. Yet, with a storm on the way—you can hear the thunder—and your brother mysteriously missing from the car, no living soul could blame a battered motorist for seeking shelter in the first available mansion—right? Maybe not, but in Mindscape's "The Uninvited" (\$49.95, for 768K IIgs) the problem is all of those other guys!—like wandering spectres, hell hounds, and some bunch of wierdos mixed up in an occult power struggle.

Designed by ICOM, the "Deja Vu" people, "Uninvited" boasts the same beautiful partially animated super-res artwork, IIgs sound high-lighting, and 'action button'-plus-windows picture-text format. Just click on an item for a description, drag it to your Inventory window to GET it, etc.. OPENing a wallet, box, or other container, produces additional (resizable) windows for viewing, using, and moving the contents. You CAN enter text—it's the only way to try out a magic phrase, for instance—but a few mouse moves will usually suffice. Indeed, a button to OPERATE one clicked item (e.g. a key) upon another (e.g. a lock) handles many actions much more expeditiously than text. Speedy, multi-position 'Save as' and 'Open' reduce the risk when testing solutions; and a 'Visible Exits' window assists mapping. Good thing!, because the mansion is big, laced with traps, and brimming with use-this-to-get-that-to-find-those... puzzles. Expect a helpful manual, two mini-diskettes, and several hours of challenging instruction in what it means to be "The Uninvited"!

Printing Out the Parm Names

According to the technical support person who answered my call at Central Point, among the most frequently requested "Copy II Plus" changes is an option to allow printout of just the names of supported programs. (One can obtain a names-only display or names-plus-parms print-out.) Why, then, as of Version 9.0, hasn't the feature been added? My guess is that the CPS people figure "Why bother? Sooner or later someone at Computist will take care of it." True enough! When updating to '9.0, I wanted to be sure that old (version 7.2) parms weren't being lost; but, I did NOT want to deal with page after page of mainly parms while checking program names. Clearly, it was time to accept the CPS challenge.

Results? Well, you need not be concerned about doing a listing just to salvage vintage parms. Only one or two obscure entries were dropped (possibly because the programs never actually existed). On the other hand, just having a hardcopy names catalog does turn out to be a great time saver. You can tell at a glance not only whether or not a product is supported, but also how its name must be entered when doing an

Auto Copy. "PNP", the quickie BASIC program listed below depends upon certain quirks of "Copy II Plus" parm files to extract the name of each product plus any parenthetical remarks relating to product version. It will supply a printout and/or a screen listing.

PARM NAMES PRINTOUT

```
10 REM PNP: PRODOS C2+ PARM NAMES
PRINTOUT
20 TEXT : HOME
50 INPUT "C20PARM0FILE0NAME0OR0
CAT0>>0" ;Q$
52 IF Q$ = "CAT" THEN PRINT CHR$
(4) "CATALOG" : GET Q$: PRINT :
GOTO 50
55 PRINT CHR$ (4) "BLOAD" Q$
",TTXT, A$1000,B$200"
57 N$ = Q$
60 F = PEEK (48859) + 256 * PEEK
(48860)
65 PRINT "LENGTH=0" ;F "0>>0" ;:
GET Q$: PRINT
100 S = 4096
110 Q = 0
150 PRINT : PRINT "PRINTOUT?0(Y/
N)0" ;: GET Q$: PRINT Q$
155 IF Q$ = "Y" THEN PRINT CHR$
(4) "PR#1" : PRINT CHR$ (27)
"L007"
160 PRINT : PRINT N$
200 FOR I = 0 TO F + 1
210 Z = PEEK (S + I)
215 IF Z = 0 THEN ZZ = PEEK (S + I
+ 1) : IF ZZ = 0 THEN 400
220 IF Z < 128 THEN Q = 1: GOTO
250
230 IF Q THEN PRINT :Q = 0
240 PRINT CHR$ (Z) ;
250 NEXT I
400 PRINT : PRINT CHR$ (4) "PR#0"
405 GOTO 20
```

Checksums

10-\$BADD	100-\$1857	220-\$C2D5
20-\$C2BF	110-\$F899	230-\$D40B
50-\$0F35	150-\$F1A1	240-\$236B
52-\$22B2	155-\$8B6A	250-\$9291
55-\$DBE0	160-\$1E5B	400-\$9D04
57-\$0BC5	200-\$18DA	405-\$E7FF
60-\$2337	210-\$4F72	
65-\$BB52	215-\$6F70	

No Foolin'

Last year's Apple II predictions were, mostly, on-target. According to a Reuters News Service release, II series shipments fell nearly 52%. Many new games have not been released in a II format, some users have defected, and a few established publications (notably dear old CALL Apple) disappeared. On the other hand, the Great Apple Dump predicted by some, turned out to be a 'Dump-ling'; net user base probably held or increased. Most product releases continue to include, eventually, a II version; and, several very attractive products are available ONLY for IIe or IIgs. In Star Trek terminology, the II series took a 'direct hit' in '89; and has come back stronger and tougher.

Which brings us to the other half of the infamous Issue #67 commentary. True, we do not see curls of smoke rising from Cupertino, circling vultures, and fat barbarians bidding for the crown. We do see lower profits, dropping stock value, and declining market share. Big Green, as in the days just prior to its last II series 'rediscovery', needs a major, attention-getting, marketing success. Some "industry analysts" have suggested a low-priced Mac; but, aside from being a contradiction in terms, IF a for-real '90's technology Cheapo Mac were offered, the first casualty would be the current high-profit-margin Mac II. A not-for-real sub-performing Cheapo would, of course, merely repeat IBM's PC Jr. fiasco.

In following through with release of GSOS 5.0, Apple demonstrates that it is not quite ready to fall on its sword. Whether Big Green has forgotten how to wield it remains to be seen. A vast market is still wide open, ripe for plucking by the first manufacturer able to tell a "PC" business machine from a genuine "Home Computer". Apple used to know the difference; and, with Spring in the air and just a bit of prompting from its II users, may be on the verge of remembering.

Next?

Expect still more games; and I'll try to scratch the surface in Languages and Sound wares.

VENDORS

Activision/Mediagenic: 3885 Bohannon Drive, Menlo Park, CA 94025 (415-329-0800)
Apple Computer: 20525 Mariani Avenue /MS 361, Cupertino, CA 95014 (405-996-1010)
Baudville: 5380 52nd Street SE, Grand Rapids, MI 49508 (616-698-0888)
Britannica Software: 345 Fourth Street, San Francisco, CA 94107 (800-572-2272, in CA call 415-546-1866)
Broderbund: 17 Paul Drive, San Rafael, CA 94903, (415-492-3500)
Central Point Software: 15220 N.W. Greenbrier Parkway #200, Beaverton, OR 97006-9937 (800-888-8199)
Data East: 470 Needles Drive, San Jose, CA 95112 (408-286-7074)

Electronic Arts: 1820 Gateway Drive, San Mateo, CA 94404 (415-571-7171)
Epyx: 600 Galveston Drive, P.O. Box 8020, Redwood City, CA 94063 (415-366-0606)
Icom: 648 S. Wheeling Road, Wheeling, IL 60090 (312-520-4440)
Mindscape: 3444 Dundee Road, Northbrook, IL 60062 (312-480-7667)
Three-Sixty Pacific: 2105 South Bascom Ave., Suite 290, Campbell, CA 95008 (408-879-9144)
Victory Software: P.O. Box 821381, Houston, TX 77282-1381 (800-232-3828, in Texas call 713-493-3232)

Jim S. Hart NC

Softkey For...

An Introduction to General Chemistry

COMPRESS

Requirements:

10 blank disk sides
 Super IOB v1.5
 NEW SWAP controller
 RWTS WORM (from Computist #61) or equivalent method to capture RWTS
 A fast DOS such as Diversi-DOS or Pronto DOS is recommended

COMPRESS' Introduction to General Chemistry is a series of ten disks that help teach and reinforce chemistry concepts taught in the classroom. Overall, the package is an excellent one. The students I have spoken to tell me that the program has helped them out with the 'grey' areas. Thankfully, the protection on the disks is not nearly as well done as the program; if it was then I still might be trying to crack the code! A cookbook section is at the end for the bottom line folks.

The Protection

Trying out the Locksmith fast copy program results in lots of inverse asterisks. This tells us that the disk at least has a format protection scheme. Using COPYA and getting a 'Disk Error' error message means the same thing. What that means is the information on the disk has been written in such a way so that normal copy programs can not read the disk.

The disk itself has code (called the RWTS, or Read/Write a Track/Sector) that can read it's own format so that is why the disk works when it is booted: the disk's DOS is using the patched RWTS to read itself. What we need to do is to find a way to alter our own normal RWTS so that it can read the perverted format on the protected disk. The easiest way is to capture the protected disk's RWTS and use the NEW SWAP controller in conjunction with Super IOB.

Back in issue #61, a program of mine called RWTS WORM was published. It's sole purpose was to 'pseudo boot' a protected disk and capture the RWTS. If you don't have that issue, I suggest you get it so you can type in the RWTS WORM program. It works on many disks and makes capturing the RWTS easy.

I thought that using RWTS WORM on the first of the ten General Chemistry disks would be sufficient. What I meant is that I thought that each of the ten disks was protected with the same format protection, which would mean that any of their RWTS's could be captured and then used on all of the disks. Wrong. I found that each of the disks had a different format alteration and thus would require it's own RWTS in order to copy it. This method of using RWTS WORM on each of the disks worked fine, albeit it was a cumbersome process.

Once all of the disks had been converted into normal DOS 3.3 (16 sector) format, I used Copy II Plus v5.5 to make sure that none of the files was on track #2, and I then copied Diversi-DOS onto each of them to speed the booting process. All of the disks seemed to work just fine.

1. Boot DOS 3.3 (preferably a fast DOS), and initialize each of the ten blank disks.

INIT HELLO

2. Load RWTS WORM into memory and capture the foreign RWTS.
BLOAD RWTS WORM,A\$9500
 (insert General Chemistry disk #1)
CALL 38144
FP

3. Insert disk with Super IOB and NEW SWAP controller, load them, and merge them.

4. Add a line to the controller.

1015 TK = 3

5. Copy General Chemistry disk #1 and answer 'NO' to the format destination disk prompt.
RUN

6. Repeat steps 2 through 5 for each of the General Chemistry disks.
 You're done!

Softkey For...

Organic Chemistry

COMPRESS

Requirements:

8 blank disk sides
 Super IOB v1.5
 RWTS WORM (from Computist #61) or equivalent method to capture RWTS

NEW SWAP controller
 Ultima IV controller from Computist #28
 A fast DOS such as Diversi-DOS or Pronto DOS is recommended

COMPRESS' Organic Chemistry is another segment in a series of programs that reinforce chemistry concepts for students, either high school or college. The program's protection schemes are very much like the ones used in the Introduction to General Chemistry disks. Refer to that article for an explanation of capturing the RWTS and what the RWTS is.

1. Boot up DOS 3.3 (preferably a fast DOS), and initialize each of the eight blank disks.

INIT HELLO

2. Bload RWTS WORM into memory and capture the foreign RWTS

BLOAD RWTS WORM,A\$9500

(insert Organic Chemistry disk #1)

CALL 38144

FP

3. Insert disk with Super IOB and NEW SWAP controller, load them, and merge them.

4. Add a line to the controller.

1015 TK = 3

5. Copy Organic Chemistry disk #1 and answer 'NO' to the format destination disk prompt
RUN

6. Repeat steps 2 through 5 for Organic Chemistry disks #3,4,5,6, and 7.

7. The protection on disks #2 and #8 is the same as the protection that is used in the game Ultima IV. When I tried the method above for copying these two disks, nothing but errors came up. Hmmmm. I looked at the disk's format with a nibble editor and discovered that the protection scheme was the same as Ultima IV's. I tried the Ultima IV controller and it worked like a charm. Install the Ultima IV controller into Super IOB and copy disks #2 and #8 onto the remaining blank two disk sides.

You're done!

Changing levels on Tetris (IIe)

I enjoy the game Tetris quite a bit. The mental game of placing blocks in the right spot is most addicting. One thing I don't like, however, is that the level you are on increases. A higher level means the blocks move faster. Faster blocks means less time to think about where you want to place the blocks. The increased speed makes Tetris into more of a reaction game instead of a thinking game. I decided to do something about that by hacking into the code. A few hours later, I came upon a section of code that increments the levels. Once this section of code is modified, the levels never increase and you can set what level you want to play at. If you've outgrown levels 1-5, just set the level to 6 and that's where you'll stay. After practicing on the modified version, play on the normal Tetris will be easier.

1. Boot up your IIgs (Rev. 01 ROMs) and get into BASIC.

2. Go into the monitor and activate the visit monitor CDA:

CALL -151

#

3. Boot up your Tetris //e disk. I used the 48K version.

4. Once you get to the main title screen, press [open-apple ctrl esc] to bring up the CDA menu. Choose the VISIT MONITOR option and you will be put into the monitor.

5. Type in the following command to find the first byte sequence that we need to change:

VD E1 73 8D 7E 78 20 E1 72<300.BFFFF

What this command does is to search the memory range 300.BFFF for the byte sequence between the reverse slashes (\). After the command is entered and the RETURN key is pressed, an address will show up. This is the address that the byte sequence was found at. For example, look at the following:

VD E1 73 8D 7E 78 20 E1 72<300.BFFFF you type this

6000: the computer responds with this

6000:A9 05 EA then you change the level to 5

You change the byte sequence:

AD E1 73 8D 7E 78 20 E1 72 original

A9 xx EA 8D 7E 78 20 E1 72 change to

where 'xx' is the level you want to play at. The level can be a number from 1 to 9 inclusive.

6. Now that you have chosen the level you want to play at, let's get rid of the code that increments the level number:

VE E7 78<300.BFFFF

Replace both occurrences with EA EA EA. (Was EE 7E 78).

7. You're done making changes. Press ctrl Y to get back to the CDA menu and then choose the QUIT option. You will be returned to the main menu you left. The starting level option doesn't do anything now, but the number of rows of bricks still does.

Marc Batchelor FL

I would like to try to address some of the questions that other readers have asked lately.

To: Fred Sheim

Re: The Observatory

I would suggest trying some variation of the controller and/or softkey for Crossword Magic 4.0 printed in Computist 64 or 66. To check to see if this might work, boot up CIA (if you have it) and use the Linguist to examine the track number on track 1 (or 2, 3, etc). The object is to see if all tracks have a track ID of 0. If this is the case, and if the sectoring is fairly normal, give one of the aforementioned controllers a try. Remember, you will still have to write a loader routine or modify theirs to accept track numbers other than 0.

To: George Cawthorne

Re: Sectors/Blocks

I would recommend purchasing Beneath Apple ProDOS. On page 3-17 is a chart that gives a cross-reference of Track, Block, Physical Sector and Dos 3.3 Sector. The book is available from Quality Software, 21610 Lassen #7, Chatsworth, CA 91311.

To: J. Richard Demonowski

Re: Boot Tracing

When you execute the command 8600G, it is SUPPOSED to boot the disk. C0E8 is a command that you type, not a prompt. When following the softkey, keep in mind that you type what appears in BOLD FACE letters.

To: W. L. Stallard

Re: RAM Disks

I would be able to answer your question better if I had an idea of what programs you are referring to. I have a //e with Ramworks III and I have had no problems with softkeys that call out use of a Ram Disk. Please state which programs you are having difficulty with and I'll try to give an explanation.

To: Jerry P. Mulder

Re: Modem Questions

I have an ADT external 2400 Baud modem, purchased through The Computer Shopper. It has all the nifty capabilities and is compatible with all modems which use the standard "AT" set. I use Proterm 2.1 (excellent program, requires //e (enhanced), //c, //gs, non-copy protected).

To: Cristian Preda

Re: F-15 Strike Eagle

Join the club. There have been 5 different softkeys published in Computist (issue 24, 30, 31, 35 and 38), as well as many issues filled with notes. I would suggest that you purchase all aforementioned back issues as well as any others to aid you in your quest.

MicroProse threw the user community a curve with the protection(s) used on that one!

To: Bill Todd

Re: L.A. Crackdown

See Computist issue #67.

To: Ken Ball

Re: Tracks and Sectors

If you own Copy II Plus 5.0 and 7.2, upgrade to 9.0. On page 95 of the Copy II Plus manual begins the discussion on the Sector Editor. I also recommend Beneath Apple DOS and Beneath Apple ProDOS (address given above). These books are an invaluable reference source and probe tracks and sectors in depth.

To: John L. Moore

Re: RAM Tests

Locksmith 6.0 has an extensive RAM exerciser. Also, there is a program that is marketed by Nikrom (I think) called Master Diagnostic //e. This also has a good ram test sequence. For the self test code, go to your local Apple Dealer and request the Apple //e Reference Manual and the Reference Manual Addendum. The addendum contains the Monitor ROM source code.

To: All

Re: Copy //+ 9.0

I purchased my upgrade to 9.0 before the holidays and it arrived in a flash. The only real problem is bugs. Specifically, the program will not quit to to Prosel (and maybe other launchers). Also, on the //gs, the program simply crashes after bouts with flashing screens and inability to quit. Central Point software told me that they will be shipping an update for these problems, so, if you are experiencing these problems, give them a call. They will put you on the "list" to receive the free update. Many kudos to Central Point for admitting that 9.0 has serious bugs. Unfortunately, the product was not adequately tested prior to release.

I would like to add my name to the list of volunteers to help fellow computists. My address is:
Marc Bachelor
6025 Coker St.
Cocoa, FL 32927

Also, thank-you volunteers at Computist. You all do a fantastic job at producing the most worthwhile magazine I have ever read. I have a couple of suggestions to make editing easier (don't we all). I would suggest that you print a small note on how to (or how NOT to) format our text files. I know I go through great pains to make sure that the title is centered (which probably irritates the person who has to remove the spaces). Or how about this, where do we put our names to make it easier for you to edit the files. Is it better to put each softkey in a separate file or use one huge file? Do you want segregation for IBM contributions? How can WE as contributors make YOUR job easier?

Boy am I glad you asked. Now it won't sound like I'm carping at the readers.

While it is more important that everyone write than that they write perfectly, it does help if they do certain things.

Put your name and address info at the start of your letter, not at the end. If you want to use a pseudonym (or handle) instead of your real name put it at the start also.

Don't use any formatting except for column type data such as code listings which should be formatted with spaces or tabs. If there is some special formatting that you want us to use, send an annotated hard copy along with your disk and we will insert the proper formatting commands for our system.

Use an editor that only puts returns at the end of a paragraph and NOT at the end of each line, unless that's the only editor that you have.

Don't use spaces to indent paragraphs, use returns instead to separate the paragraphs. Use as many returns as you like to separate paragraphs or headings to make them more readable (I do all the time). Our translate program strips out multiple returns when we send the file to PageMaker. And always, always, always, send BASIC programs as BASIC programs. Just SAVE them to the disk. Don't send them as text files, especially as formatted text files. It's a horrible task to have to carefully remove the formatting and then exec the resultant text file, in order to change it back to a BASIC program.

The same goes for Binary programs. Just BSAVE them to the disk. ...RDEXed

Comments on IBM Ragging

A few issues ago, I read a readers input that read something close to the following: "Imagine having to change ROMs to change your DOS". That comment, like other false statements were directed toward the IBM and it's famous (infamous) MS-DOS. As background information, let me say that I have owned my Apple //e since 1984, and have been programming the Apple // series since my freshman year in High-school (quite a long time!). At work, I use a Televideo 970, as an interface to a Honeywell, an Apollo DN-3000 and an IBM PS/2. Now, for all those who proclaim that the IBM and MS-DOS is such a user-hateful environment, consider this:

- 1) Have YOU ever used one?
- 2) Do you know ANYTHING about MS-DOS, its command structure, and its power?
- 3) Do you really believe that an 8 bit machine is the only good machine available?

Now, imagine this:

1) The ability to create what is called a "PATH" that sets the hierarchy for locating files in multiple directories (something that I would KILL for in ProDOS).

2) Imagine easily creating your own commands without needing to patch RAM, the operating system or anything else. Just save the program into a directory that is in your path, and WHAM... instant new command.

3) Imagine not needing the "BRUN" command. Just type the name of the application and watch the program start.

3) Imagine the speed of flying through calculations or sorting at speeds of 33 Mhz.

4) Imagine more programming languages than you can learn.

5) Imagine true multi-tasking and windowing.

Now I'm certainly not saying that IBM has all pluses and no minuses. But, before you condemn a product or technology, please be sure of your information. Incidentally, addressing the comment mentioned earlier, MS-DOS and PC-DOS (practically identical) reside as hidden files on the floppy disk. Tandy corporation (Radio Shack) does produce an IBM clone that does have DOS in ROM, but that is the users OPTION, not a mandate and it is certainly not the standard.

Now for my question. I have been working steadily on Where in the USA is Carmen Sandiego (Carmen) and Wings of Fury (WOF), both by Broderbund. On Carmen, "sector" 0 on any given track begins with 9A D5 XX and all other "sectors" on the track just have D5 XX XX. Both disks appear to be formatted in 4 & 4 encoding. I would love for one of the more enterprising programmers in the group to come up with a standard 4 & 4 copy program to use to crack some of these disks. For 4 & 4 encoding, a byte is read from the disk, ROL'd, and AND'd with the next byte from the disk to produce one data byte. This is not exactly a rough process (considering DOS 3.3's translation tables), but I just don't have the time to develop such a sophisticated copy routine. Maybe this will give someone with more time the inspiration (or challenge) to create such a program. The translation of 4 & 4 is as follows, X REG = slot * 60.

```
A0 00 LDY #00
START BD 8C C0 LDA $C08C,X
10 FB BPL START
2A ROL
85 26 STA TEMP
NEXT BD 8C C0 LDA $C08C,X
10 FB BPL NEXT
```

```
25 26 AND TEMP
99 00 20 STA DATA,Y
C8 INY
D0 XX BNE START
```

Of course this doesn't take headers, trailers or sync fields into account, but this is just an idea. That's all I have for now.

A reader review of the

Trac Card

Midwest Microsystems
10308 Metcalf, Suite 355
Overland Park, KS. 66212
\$159.00

Overview

Trac Card (TC) is a card that can be placed into any slot from 1 to 7 (including slot 3 in a //e). The purpose of TC is to "follow" the tracks accessed by any program and store the track numbers into it's "on-board" memory. The card can store up to 2000 track numbers, which are divided into groups of "booted" disks. The card accurately stores 1/4, 1/2 and 3/4 track accesses as well. Also included in the package is DOS 3.3 and ProDOS versions of the support software.

I have been using TC for well over a year. It is almost as indispensable as my Senior Prom in cracking disks. The typical usage would be as follows:

- 1) Place disk to boot into drive 1 and boot it.
- 2) When the disk access stops, take program out of disk 1 and replace with support utilities disk.
- 3) Press ctrl open-apple reset (IIe/IIgs) or ctrl reset (II+) to boot up support utilities.
- 4) At menu press "D" (to display quick), and presto.... a Hex Boot Trace of your program.

The term "boot-trace" is used here differently than it is normally used in this magazine. In this case, boot-trace refers to a list of tracks accessed by the program being traced in the order that the program accessed them.

Installation

Installing TC was as simple as moving a jumper block and popping it into an available slot. There are no cables or wires to get into the way.

Support Software

The support software is extremely flexible, allowing you to set things up by individual need. The main menu is as follows:

```
TRAC CARD (C)1986 CGL MICROSYSTEMS
DISKS BOOTED=x
(1) DECIMAL TRACKS IN ORDER
(2) HEX TRACKS IN ORDER
(3) DECIMAL BOOT TRACE
(4) HEX BOOT TRACE
(5) PRINT DECIMAL TRACKS IN ORDER
(6) PRINT HEX TRACKS IN ORDER
(7) PRINT DECIMAL BOOT TRACE
(8) PRINT HEX BOOT TRACE
(A) ADD NAMES FOR PRINTOUT
(D) DISPLAY QUICK
(P) PRINT QUICK
(S) SET SLOTS / DISPLAY
(E) END
SELECT ONE OF THE ABOVE
DRIVE=x TRAC=x PRINTER=x DISPLAY=40
```

I think most of the menu options are self explanatory. Option "S" gives another menu, which allows one to set up disk drive slot, TC slot, printer slot, display (40/80 Columns), and Display Quick/Print Quick setup.

Usability

I was very impressed with everything about this product (which is really a shock!). This is about the only product that I've encountered that performs at or above expected levels. The support software is fast, and friendly and the output is easy to read and understand. Additionally, I have encountered no software that even acknowledges it's presence, therefore, compatibility is not even an issue. It is totally transparent to all other software and hardware.

Shortcomings

The Trac Card (at present) only functions with 5.25" drives. I hope that an update for 3.5" drive will be forthcoming. If however, all you own is a 3.5" drive, you might consider contacting CGL Microsystems to see if a 3.5" version is being considered.

Conclusion

TC is a must for any serious Apple // user, and is compatible with the //+, //e, and //gs. TC is an excellent investment at \$159.00 (plus \$3.00 S & H), and in a short time will be a "must have" in your cracking kit.

Mark J. Ruskin NY

IIgs Softkey for...

Graphics Studio

Accolade

I've had Graphics Studio for quite a while now and use it more than Paint Works Gold or Deluxe Paint II. Once I got used to the different interface I found I liked it better than the other

paint programs. You don't have to hold the mouse down when opening a menu item as with almost all the other programs I've got, also, like Paint Works Gold you can scroll through your fonts. Since I've got a hard drive with over 90 fonts this is very useful to me.

Well anyway, back to the reason I'm writing. When I got my Graphics Studio disk I pulled out my issues of Computist and looked up the softkey for it. I found it in COMPUTIST #54 by Brian Troha. The deprotection he had as being on Block \$31, starting at Byte \$14 wasn't there. So I ran a search with Copy II plus and found it on the same Block but at a different start location, no problem. But there was a problem as mentioned by Brian in Computist #62 page 25 where he wrote a fix for it. The only thing was I must have a different version than Brian because the protection scheme again wasn't where it should be. Brian's Softkey was:

Blk	Byte(s)	From	To
\$31	\$4C	D0 42	80 35
\$98	\$1A8	A2 20	80 2C

I did find the protection using both issues of Computist, but on my version the protection was:

Blk	Byte(s)	From	To
\$42	\$23	D0 42	80 35
\$98	\$1DF	A2 20	80 2C

This worked just fine and now I've got Graphics Studio on my hard drive and don't have to pick a tool before I load a picture without crashing.

By the way the two strings to search for are (hex) D042 AD33 0038, and (hex) A220 A001.

The newspaper type issue (#66) did have one thing missing that I wish you would put back in. That's the break down of the back issues. It's very helpful for looking up softkeys, features, and APT's. I know it takes up a lot of room, but maybe you could at least print the present year's back issues (ie. Jan. 89 to present).

Keep up the great work. I look forward to every issue and also learn from every issue.

Edison NE

Softkey for...

Dig Dug

Datasoft

Requirements:
Empty Slave Disk
128K Apple IIe

Thanks to Zorro in #58 and Brian Walker in #63 for the method used here, and Mark Harris #55 for doing all the dirty work.

1. Boot a DOS 3.3 slave disk.
2. Enter the monitor and setup to boot into auxiliary memory.

CALL-151 Enter Monitor
0:8D 03 C0 8D 05 C0 4C 00 C6

3. Insert Dig Dug disk and boot it.
0G

4. When the drive stops you should see the Hi-Res screen. (Probably filled with garbage) Press the spacebar, it should switch to the text screen. (Also probably filled with trash.)

5. Put the slave in the drive and press openapple ctrl reset.

6. After the drive stops, reenter the monitor and set up the auxiliary memory move routine.

CALL-151
300:18 4C 11 C3
3F8:4C 00 03

7. Move memory a little and save it.

2000<2000.4000 CTRL-Y
BSAVE DG.LOGO,AS2000,LS2000
4000<8000.BF00 ctrl Y
BSAVE DG.OBJ,AS4000,LS4000

8. Do step 2 through 4 again. Press J then the spacebar. (You should now here music.)

9. Do step 5 and 6 again.

1C00<0.400 ctrl Y
BSAVE DG.P0-P4,AS1C00,LS400

10. Enter the hex dump and save it.

DG.MOVER

```
1BB5: A0 00 A2 $7D16
1BB8: 00 BD 00 40 9D 00 80 E8 SCA4B
1BC0: D0 F7 EE BB 1B EE BE 1B $FA98
1BC8: C8 C0 40 D0 EA A0 00 A2 $DD99
1BD0: 00 BD 00 1C 9D 00 00 E8 $3602
1BD8: D0 F7 EE D3 1B EE D6 1B $4ACD
1BE0: C8 C0 08 D0 EA A0 00 A2 $8914
1BE8: 00 BD 00 20 9D 00 40 E8 $FED9
1BF0: D0 F7 EE EB 1B EE EE 1B $56F2
1BF8: C8 C0 20 D0 EA 4C 00 80 $130C
BSAVE DG.MOVER,AS1BB5,LS4B
```

Now BLOAD the first three files then BRUN DG.MOVER to test DIG DUG. If it runs fine, go on to the next section.

Turn Dig Dug into one File

BLOAD DG.MOVER,AS1BB5
BLOAD DG.P0-P4,AS1C00 or DG.P0-P7,
BLOAD DG.LOGO,AS2000 AS1C00 for Mark's
BLOAD DG.OBJ,AS4000

If your using my DG.MOVER jump to the BSAVE at the end. If using Mark's DG.MOVER do the following...

CALL-151
1BC3:BB 1B

1BC6:BE 1B
1BD2:00 1C
1BDB:D3 1B
1BDE:D6 1B
1BF3:EB 1B
1BF6:EE 1B
BSAVE DIG DUG, A\$1BB5, L\$644B

Dig Dug is now a 102 sector BRUNable file.

Richard Brooks **AK**

To Brent Michalski, Alex Vogt, and others. The best price that I have found on Beneath Apple DOS and Beneath Apple ProDOS is from Silicon Express (1-614-927-9555) for \$12.95 each plus approximately \$4.00 for shipping, depending on where you live, for a total cost of about \$30.00. About a \$10.00 savings over retail (could make a nice donation to Computist).

To Stephen M. Caraco: ref. #65 pg 8 — From Copy II+ V7.2 — Toy Shop

T0, Sector Copy
T5-T1B, Sector Copy
T1-T4, Sync, Keep, 0E=9A, 0F=D5, 10=9B, 55=03

T1C-T22, Sync, Keep,
T21.25, Sync, Keep
I'm not sure if this is the same pram as from V7.4 but maybe it will work for you.

To C. E. Chuck Garrett: As many can tell you, the DOS 3.3 System Master is no longer available from Apple. The best way to obtain a copy is from other apple users in your area, i.e. checking with a local user group is the best place to start. You could possibly check with a local Apple Dealer, they may have one around in their back storeroom collecting dust. If all else fails I'm sure a lot of the subscribers to Computist that have the System Master would make a copy for you if you would send them a blank disk and postage stamps to return it to you. Also COPYA is one of the programs on the System Master.

At this time I would like to say that I concur with Edward Teach. Lengthy explanations are great, especially for newbie's like myself who at sometime in the near future hope to be making the same kind of contributions. I would like to praise all the writers who go out of their way to explain in great detail how they go about accomplishing the deprotecting of their software.

I would also like to say that I like your idea about the Computist Bulletin Board, I have thought of getting a modem before but did not have a good enough reason to acquire one. This could be just the right reason I've been waiting for.

One thing that may help now is the fact that A+ has been incorporated into Incider should cut your monthly advertising cost some. One suggestion that might help would be to start alternating months between the magazines. One month in Nibble and one month in Incider, this alone could cut your advertising cost in half.

Many years ago I saw an advertisement in A+ or Incider from a place called Pirates Harbor. They were offering software on deprotection techniques. I was wondering if any one knows if they are still in business or if the software is still available through some other supply

I'd be careful of Pirates Harbor, if they're still around. They don't deliver on their promises. I won't say that they ripped off a lot of people but I know of some people who got burned. RDEXed

Jeff Root **CA**

Lifting the Lid on COPYA

You see it time and time again, "Load COPYA. Hit Control-C. Delete line 70 and patch DOS as follows..." I have followed such instructions blindly, only suspecting what the various "B942:18" were doing to my disk. Worse, there were some really neat patches that allowed you to copy from a certain track on, or from track zero to a certain track, and they all looked like Greek to me. I finally decided not to live in ignorance anymore. I was going to disassemble COPYA.

Last issue (#66) had an article in it about COPYA. Scott Simon had decided to automate his patches, and gave instructions on how to patch the Applesoft portion of COPYA to achieve this. This article had much good info in it, particularly for a machine language only programmer like me. I am at best a barely literate BASIC programmer, so I had trouble adding patches to the Applesoft code. The article by Scott Simon was useful. However, it does hold one or two inaccuracies. The first is in Scott's HELLO program, in line 70. He BLOAD's COPY.OBJ0 and adds a REMARK that it loads into \$2C0. As you will see, this is not true. I don't mean to pick on Scott, as several other articles in past issues have made the same mistake; one even patched over a section of COPY.OBJ0! Also in the HELLO program, Scott should have a line "105 PRINT D\$ "OPEN CMD\$"" and "106 PRINT D\$ "DELETE CMD\$""; otherwise the previous text file will not empty and could add garbage to the current file. While you could avoid all this by using the HELLO program only once, or make sure you never modify the program without taking this into account, I opt for adding two lines to HELLO.

Scott also had a problem with DOS being disconnected while running his modified COPYA. Well, I have not tried to trace the code to see where he went wrong, but I suspect that he is CALLING COPY.OBJ0 incorrectly. As you will see, COPY.OBJ0 continually disconnects and reconnects DOS. Perhaps Scott can track down the source of his problems with the information provided here. If so, please send in your corrections so we can use your Super COPYA 1.1 easier.

What I will do here is to give a line-by-line and byte-by-byte commentary of COPYA and COPY.OBJ0. As much as COPYA is used in COMPUTIST, it is about time someone cracked it! My goals are to produce useful comments so that you can understand COPYA, to show a uniform way to add patches and extensions to COPYA, and to provide several ready-to-use additions to make COPYA easier for COMPUTISTS to use.

First, we'll take a look at the BASIC program COPYA. I will show how it works, how to capture your own source code for modification, and a simple interface between COPYA and your own BASIC code. After that, we can look at the machine code of COPY.OBJ0, then on to some heavy mods.

I figured it shouldn't be too hard, as most of the code was in BASIC. Well, that is almost true. Having so much in BASIC made cracking the machine code easier, but COPYA is not as straightforward as it seems. First, the Apple people put some things into the program for the express purpose of making it harder to capture and modify. So the program we've all been using against those locked disks is itself protected! Also, there are some comments that are just plain wrong. I don't know if this was deliberate, or just a side-effect of 'corporate computing' (my term for what gets published as a result of your boss saying, "I don't care if it's not debugged; it works pretty good, so ship it!") but we need to clear this up.

Here then, are my line-by-line comments for COPYA:

COPYA Script

This script is for COPYA, as distributed by Apple Computer, Inc. Copyright date is 1980. The line numbers refer to the original program line numbers.

Notes: In the scripts for COPYA and COPY.OBJ0, I have used two names for the copied disk and the copied-to disk. The copied disk is called Original, and Master. The copied-to disk is called Duplicate, and Slave. This nomenclature derives from the disparity in the Apple code. The COPYA program uses 'Original' and 'Duplicate' for all prompts, but it uses 'Master' and 'Slave' for those portions of the program which interact with the machine code. I have used the 'Master'/'Slave' designations when writing the script for COPY.OBJ0.

Line Number Function
0 Put the Apple into TEXT mode. Also used to make capturing the listing more difficult.
10-60 Program title and Copyright notice. All REM statements.

70 BLOADCOPY.OBJ0. Note that while comments say this is at \$2C0, we really load at \$2A0.

80 Put title on screen.
90 Calls \$2C0. This is the initialization routine. See COPY.OBJ0 Script for details. Then sets CS to the current slot.

100-110 Sets up the buffer start and end. Both Buffstart and Buffend are on even page boundaries. Buffer start is the page byte of the bottom of free memory +1 (to avoid overwriting a partially used page). Buffer end is the page byte of the top of free memory -1 (in case of partially used page). See Applesoft memory map for details.

130 Gets the original slot and drive numbers from the user. Default slot is CS and default drive is master drive loaded by the call to \$2C0. MS (Master Slot) and MD (Master Drive) are set per user input.

132 Get the duplicate slot and drive from user. Default slot is the same slot as MS, default drive is opposite drive from MD. Note the clever use of modulo 3 arithmetic to arrive at drive numbers.

160 Set FT (First Time) to 0. This indicates that we need to init the duplicate disk. (Ref line 230).

163 Erases any old messages from a previous execution of this program. (E.g.; a second copy of the same disk, or exit and return via RUN).

165 Stash slot and drive info into page 2 variable table (\$2C9 - \$2D6).

170 Set WNDTOP to below the slot and drive info, and HOME the cursor. The program uses HTAB and VTAB to write messages above the scroll window; the space below is used to pass error messages, etc. to user.

175 Home the cursor and mess with the BASIC error code.

180 Prompt for start of copy. This allows the user to insert the disks in the appropriate drives.

185 Home the cursor.

190 Allow the user to insert the original disk if a one drive copy is being performed.

195 Put the "READING" message up on screen for user info.

200 Call the Read A Hunk machine language subroutine at \$2C3.

210 Check the return code. If an error exists (code = 2) then goto 280 and put the "UNABLE TO READ" message on screen, then exit.

225 Erase "READING" message. Check return code and goto 290 if copy is finished. (Return code = 1).

230 If this is NOT the first time through (FT = 1), then goto 255 to write memory to the duplicate drive.

240 Prompt user to insert the duplicate disk, if we're doing a single drive copy.

245 Set ONERR to our handler, 275.

246 Send message that we're initializing the duplicate disk.

250 Use DOS to INIT as "XXX". Set FT to 1. (Ref line 230). This also serves to certify the disk, so we only need to handle write errors here.

251 Erase the "INITIALIZING" message.

252 Disable the ONERR pointer and mess with the APPLESOFT error number again.

253 Branch around the "INSERT DUPLICATE" prompt, as we did that as part of the init. The message is in line 255 so that when we drop into the write routine the next time, we allow the user to swap disks.

255 Prompt user to insert the duplicate disk. We enter here on all but the first time through.

256 Tell the user we're writing to the duplicate.

260 Call the Write A Hunk machine code subroutine at \$2C6 to write memory to the duplicate disk.

265 Erase the "WRITING" message.

270 Check the return code and do more reads if no error exists.

275 Start of INIT error handler. Test if ctrl C has been pressed. If true, the program will exit with a "Break in Line..." message.

277 Tell user we're unable to write to the duplicate. Exit through line 290.

280 Tell user we're unable to read the original disk.

290 Ask user if he wants to try again. This allows the user to correct open drive doors, write protect tabs, etc. without restarting the program.

295 If user elects to try again, start over at line 160.

300 If user types something besides 'Y' or 'N', then branch to line 290.

305 Reset screen. Home the cursor. Call the machine language routine at \$2A0 to fix DOS keywords. Delete line 70; this is nice if you want to rerun the program and not have it reload the COPY.OBJ0 file. End the program. This is the normal exit point from the program.

310 Gets slot number from user. Original or Duplicate is determined by contents of IS. Allowed values are 1 to 7. Sub 350 puts the "DEFAULT=" text on screen. N is the default value. SUB 330 gets a key and does a range check to see if it is within the values set above. K=K-176 shifts the input ASCII down to hex. This sub returns with the selected slot in N.

320 Get the drive from the user. Very similar to the above sub 310.

330 Reads the keyboard. If a key is .not. pressed, then we repeat line 330. The rest of this line is a mistake. The programmer evidently thought that the logic would drop through to the rest of line 330 if a key was pressed. However, the way that Applesoft handles the 'IF ... THEN linenum' statement is that if true, we go to the linenum. If not true, then we go to the next line; in this case we branch to 335. So, NONE of the code after the 'THEN 330' is ever executed. Ref. the Applesoft Tutorial.

335 Reset the keyboard strobe. If the key pressed is a carriage return character, goto line 340 and accept the default.

336 Shift the key value down to the 00-09 range by subtracting the ASCII value of '0'. Test this to see if we are in the range allowed. If not, get another key.

337 We get here if we have a key pressed that is in the range allowed. So set N to this value and drop through to 340.

340 Print our selection and return to caller.

350 Print "Default = " in inverse, then return to caller.

360 If Original and Duplicate are in different slots, return.

370 If Original and Duplicate are in different drives, return.

380 Tell the user to insert disk and get return key to continue. This is where we handle the one drive copy problem. The prompt becomes transparent to the program.

385 Home the cursor to clear the "Insert disk" prompt.

390 Return to caller.

Now that you know what everything does, you'll want to capture your own code for modification. This means turning the Applesoft program into a text file. Apple has generously given us the procedure for this in "The DOS Manual". You simply add program lines to the beginning of COPYA that open a text file and list the 'real' program lines into it. Something like the following should work:

MAKE SOURCE

1000 REM

```
1001 REM TO USE: LOAD COPYA
1002 REM EXEC CAPTURE.LIST
1003 REM RUN 1000
1004 REM
1011 PRINT CHR$(4); "OPEN COPYA"
SOURCE,D2"
1012 PRINT CHR$(4); "WRITE COPYA"
SOURCE"
1013 POKE 33,30
1014 LIST 1,999: REM LIST PROGRAM.
1015 PRINT CHR$(4); "CLOSE COPYA"
SOURCE"
1016 TEXT: END"
1017 REM
1018 REM NOTE THAT WE DON'T TRY TO
1019 REM DELETE THE FILE FIRST, AS
1020 REM WE RUN THIS ONLY ONCE.
1021 REM
```

Checksums

1000-\$356B	1012-\$008E	1018-\$8BDC
1001-\$7F4B	1013-\$28AC	1019-\$AC4E
1002-\$826E	1014-\$AAE1	1020-\$53DC
1003-\$DE2F	1015-\$3E65	1021-\$74F9
1004-\$668C	1016-\$B463	
1011-\$BBA1	1017-\$1FF6	

One nasty has to be dealt with here. Line 0. If you list from line 0, you will get both the program and your capture lines. OK, so you are going to just delete these with your word processor, but why not just list lines 10 through the end? Your Applesoft manual will tell you that the purpose of a line #0 is to do a '...to the end' command. Well, the author of COPYA used this as a real line #, so now we are back to square one. I suspect that the only real purpose for line #0 was to make it harder(?) to capture the source code. Line 0 is morally bankrupt: trash it by capturing only lines 10-999 and add the "TEXT" to line 80. While the program is in the machine, just delete line 0 with a "J0<RETURN>", and modify line 80 as above. Then do the capture.

Based on the above disassembly, you might also want to delete everything on line 330 after the 'THEN 330' code. Alternatively, you might want to rework the logic, so that line 330 will work as a stand-alone subroutine. I think that was the original authors intention, and it might come in handy elsewhere also.

And while we are at it, we might as well clear up something that has always bothered me. Fix the comment in line 70 to read, "REM AS2A0". Now you know where it really loads COPY.OBJ0.

COPY.OBJ0 Script

This is the script for COPY.OBJ0, a collection of machine language modules which speed up COPYA. There are basically three routines: Clean-Up lives at \$2A0 and serves to fix some stuff that COPYA trashes; Read-A-Hunk lives at \$2F7 and serves to read as many sectors as will fit into free memory; Write-A-Hunk lives at \$317 and serves to write all of free memory onto the duplicate disk. COPY.OBJ0 is very compact code, using every byte as many times as possible. Hidden in this code is the ability to patch/edit any sector during the copy process.

Address Function

2A0 Clean-Up code. Fixes the DOS keyword values table (at \$AA66 for 48K DOS.) Puts the Master Slot and Master Drive values into the table.

2BD Unknown stash, never referenced. Possibly garbage.

2C0 Main entry from COPYA. Holds a jump table to the 3 segments of code: Get Pointers at \$2D7, Read a Hunk at \$2F7, Write a Hunk at \$317.

2C9-2D6 Main variables and constants:
2C9 Return code; 01 = DONE, 02 = ERROR.

2CA Last volume accessed. This is set to original disk.

2CB Page byte of bottom of free memory.

2CC Page byte of top of free memory.

2CD Slave slot * 16.

2CE Master slot * 16.

2CF Slave drive number.

2D0 Master drive number.

2D1 Slave track number.

2D2 Master track number.

2D3 Slave sector number.

2D4 Master sector number.

2D5 Slave command code. Always WRITE.

2D6 Master command code. Always READ.

2D7-2F6 Get Pointers routine. Sets IOB pointers in \$3C, \$3D. Sets track and sector in 2D1-2D4 to \$FF. Gets slot*16 from DOS IOB and stashes it in the Master slot (2CE). Gets drive from DOS IOB and stashes in Master drive (2D0). Restores registers and back to COPYA.

2F7-316 Read-a-Hunk entry. Sets IOB pointer. Selects "Master", track number and fixes if first time through. Then it goes to Fill Memory via \$309. Tests for Read Error upon return, updates IOB volume # in preparation for INIT, and exits via \$325.

317-320 Write-a-Hunk entry. Sets IOB pointer, selects "Slave", and fills disk from memory. Tests for error upon return and exits via \$325.

321-324 Sets error code (=02) and exits via \$331.

325-330 Normal exit. Stashes current slot*16 and drive in \$0F, \$10. Sets return code to 00.

331-337 Common exit. All roads lead here. Saves return code in \$2C9, restores registers, and

axis back to COPYA.
338 - 347 Sets \$3C6 to 01 for Master Disk and to 00 for Slave Disk. Gets page byte of bottom of free memory (less one). Stashes LOMEM in \$3CA. Set up IOB to match any volume.

348 - 363 Get next available page of memory and check to see we don't hit top of free memory. Get next sector to read, test to see if it is 0-15. If not, move to next track less than \$23. If we are at track \$23, then clear the carry and return.

364 - 3A1 Set sector to 15 if new track. Set IOB to use our page byte (\$3CA) and 00 for the buffer address. Set slot*16 and drive in IOB. Set track number and sector number along with RWTS command code (always READ or WRITE) in IOB. Set IOB address using page 3 hook. Reset IOB pointer in \$3D, and \$3C. Go to \$3A2 for normal exit. If error return with carry set.

3A2 - 3AA Get Master/Slave indicator. Drop down to next sector. Branch to \$348 to read/write next sector.

3AB - 3BB Stash registers, get IOB address. Set \$3C, \$3D to point to IOB. Return.

3BC - 3C5 Restore registers, return.

3C6 - 3CA Stash:

3C6 - Holds 0 for the Slave disk; 1 for the Master disk.

3C7 ASAVE.

3C8 XSAVE.

3C9 YSAVE.

3CA Page byte of available buffer in memory.

You will note some places in my comments where I am not altogether clear. This is because I don't know everything, and I made the best guess I could. Those of you out there who can make something in here clearer, write in and help me.

Well, the last thing I want to leave you with is a simple extension to COPYA that allows you to add your own routines. There is one rule for modifying COPYA; do what you like, but don't mess with the Apple code. Instead, let's use a GOSUB to attach a custom DOS patcher to the main code.

COPYR MOD

LOAD COPYA

75 TEXT:GOSUB 1000

custom functions
deletes line zero

0 Add the following lines:

1000 REM * CUSTOM FUNCTION

SUBROUTINE

1010 REM *

1100 INPUT

"WOULD YOU LIKE TO PATCH DOS ?"

;QS

1110 IF LEFT\$(QS,1) = "N" THEN

RETURN

1120 IF LEFT\$(QS,1) < > "Y" THEN

1100

1130 INPUT

"IGNORE DOS READ ERRORS?" ;QS

1140 IF LEFT\$(QS,1) = "N" THEN

1200: REM CONTINUE TO NEXT

PATCH

1150 IF LEFT\$(QS,1) < > "Y" THEN

1130

1160 POKE 47426,24

1200 INPUT "IGNORE RWTS ERRORS?"

;QS

1210 IF LEFT\$(QS,1) = "N" THEN

1300

1220 IF LEFT\$(QS,1) < > "Y" THEN

1200

1230 POKE 48712,24

1300 INPUT

"MODIFY ADDRESS MARKS?" ;QS

1310 IF LEFT\$(QS,1) = "N" THEN

1400

1320 IF LEFT\$(QS,1) < > "Y" THEN

1300

1330 GOSUB 2000: REM MOD OR IGNORE

ADDRESS

1400 INPUT

"IGNORE DATA CHECKSUM?" ;QS

1410 IF LEFT\$(QS,1) = "N" THEN

1500

1420 IF LEFT\$(QS,1) < > "Y" THEN

1410

1430 POKE 47406,0

1500 RETURN

2000 REM *

2010 REM * Modify/Ignore Address

Marks

2020 REM * Careful, this kills!

2030 REM *

2100 INPUT

"MODIFY OR IGNORE FIRST BYTE

? (M/I):" ;QS

2110 IF LEFT\$(QS,1) = "M" THEN

2200

2120 IF LEFT\$(QS,1) < > "I" THEN

2100

2130 POKE 47447,0: REM IGNORE 1ST

BYTE.

2140 GOTO 2300: REM M OR I 2ND

BYTE

2200 INPUT

"WHAT DECIMAL VALUE FOR BYTE ?"

;D

2210 POKE 47445,D: REM READ MARKER

ONLY

2300 INPUT

"MODIFY OR IGNORE THIRD

BYTE?" (M/I):" ;QS

2310 IF LEFT\$(QS,1) = "M" THEN

2400

2320 IF LEFT\$(QS,1) < > "I" THEN

2300

2330 POKE 47457,0

2340 GOTO 2500

2400 INPUT

"WHAT DECIMAL VALUE FOR BYTE ?"

;D

2410 POKE 47455,D: REM READ MARK

ONLY

2500 INPUT

"MODIFY OR IGNORE THIRD BYTE

? (M/I):" ;QS

2510 IF LEFT\$(QS,1) = "M" THEN

2600

2520 IF LEFT\$(QS,1) < > "I" THEN

2500

2530 POKE 47468,0

2540 GOTO 2700

2600 INPUT

"WHAT DECIMAL VALUE FOR BYTE ?"

;D

2610 POKE 47466,D: REM READ MARK

ONLY

2700 RETURN

Checksums

1000-\$356B	1330-\$70BC	2210-\$0912
1010-\$1270	1400-\$AAF1	2300-\$DEB8
1100-\$9962	1410-\$36D4	2310-\$3CD8
1110-\$32B6	1420-\$2528	2320-\$63A3
1120-\$BD59	1430-\$8CEF	2330-\$11E5
1130-\$77A0	1500-\$FAE9	2340-\$7CCD
1140-\$6FBF	2000-\$62D6	2400-\$A3CD
1150-\$CBE7	2010-\$4941	2410-\$DE15
1160-\$D5E4	2020-\$228B	2500-\$0A2B
1200-\$D9E6	2030-\$B0C9	2510-\$6B8B
1210-\$1F27	2100-\$F4D5	2520-\$2C2B
1220-\$4B73	2110-\$4FE2	2530-\$EEFF
1230-\$EEC3	2120-\$6D8E	2540-\$CE57
1300-\$4ABD	2130-\$990A	2600-\$F59F
1310-\$3F58	2140-\$1CCC	2610-\$85E2
1320-\$DD40	2200-\$273D	2700-\$D259

Save this version of COPYA as "COPYR".

One reason that I like this technique is that the Apple code seems to be sloppily written and very defensive. Instead of fixing the faulty logic in line 330, the programmer just added in more code to cover the bug. So, if we modify it too much, one of these hidden bugs is liable to find us. Go ahead and rewrite the custom functions subroutines any way you like; if you find a real gem, write in and let us know.

For those of you who would like it, I am putting the complete set of files on disk. Plus one or two 'extras' thrown in, including an automatic COPYA patcher that implements all the extensions here plus fixes some bad Apple code. Watch COMPUTIST for details on when this will be available.

To A. Evans: You wanted a non-NMI crack for FROGGER. Well, here is a 98% crack for you. I use Main Street Publishing version of the Sierra On-Line FROGGER. I did this one a while ago so I can't remember exactly how I found the DOS patches to make; I am sure it was the result of trial and error.

Softkey for...

Frogger

Sierra OnLine

Requirements:

FROGGER disk

FID from System Master

ProntoDOS from Beagle Bros.

Procedure

1. INIT a blank disk using ProntoDOS. This makes the resulting copy boot faster and as Frogger sits from \$1FFD to \$9900, it prevents interference with DOS.

2. Now boot the system master disk with FID on it.

3. Enter the monitor.

CALL-151.

4. Patch DOS to ignore the first address byte (D5):

B954:29 00

5. Patch DOS to ignore the address checksum:

B98A:00

6. Return to BASIC.

ctrl C

7. Startup FID.

BRUN FID

8. For the filename type "FROGGER".

9. Wait. This process takes much longer than expected, as we have almost crippled DOS.

10. Make your boot program run FROGGER using the standard (boring) method:

10 PRINT CHR\$(4);"BRUN FROGGER"

A couple of notes. First, I call this a 98% crack because I get erratic results with the status line. However I have played my version through several screens without problems. Second, you may conserve disk space by ignoring the title screen. The code at \$1FFD is a JMP \$4000, which takes you over the title screen to the real code. You could modify this so that the program BRUN's at \$4000 using a sector editor. Note that you would

lose the prompts on the title screen. If this is only your copy then this is not a problem. However your should fix this to give some kind of prompt for the user. I just left it alone and used ProntoDOS to speed up the loading. If you try running FROGGER with a regular DOS, you will see how slow it really is.

I am working on fixing the Joystick code so that it will work better. Right now, you have to move the Joystick back to the center before you can move again; this is stupid. If I get it working, I will write in and let COMPUTISTS in on what and how.

While I am on the subject of COMPUTIST, let me give the editor the feedback he has been asking for. First, I am surprised that he is advertising lower subscription rates. The old rate was \$32 per 12 issues. The new rate is \$24 per 8 issues. Yes the total price of 1 years worth of COMPUTISTS is lower; but if you do the math you will discover that the price of the magazine has gone UP from \$2.67 per issue to \$3.00 per issue. Remember, Mr. Haight, we DO own computers!

I have not let, nor do I intend to let, my subscription lapse over this example of creative advertising. The magazine is well worth reading. However I would like to make two suggestions to the COMPUTIST readers about how they can contribute to increasing circulation.

First; if you have disassembled a program or even a portion of that program, let us see the disassembly. Now while this can get tricky with copyright and all, using what Don Lancaster calls a SCRIPT to present your disassembly gets around that very nicely. See my COPYA article (which should be in this very issue) for an example. If we all have a piece of a program, maybe together we have the whole program. Obviously, this allows us to modify and enhance that program as a group. Let's see your code! This would help increase circulation also by bringing in readers who are interested in intermediate and advanced programming, as they could watch a Paul Lutus or Bob Lissner do his thing. Note that there is a large group of people out there who have lost "Apple Assembly Line" magazine and are looking for a substitute.

Second; get with the times. I have seen pleas for help from those newer users who don't have DOS 3.3 or any of the tools that came with it. I am one of those who still thinks that DOS 3.3 is a superior Operating System, but let's face it: DOS is dead! These new users have just as much need for cracking tools as the older DOS users, but no one has given them the info they need. Comments like "if you don't have DOS - get it" will only drive subscribers away. As Tom Weishaar is fond of pointing out, more Apples were sold AFTER the advent of ProDOS than before. That means that more potential subscribers are using ProDOS than DOS. Don't shut them out by using COPYA or MUFFIN. I know that I just wrote an article on COPYA, but note that it has something in it for programmers as well as DOS users. The place to start is by thoroughly knowing and using ProDOS tools.

So let's combine both of my suggestions and disassemble the ProDOS Filer and Utilities right here in this magazine and watch the new people flock to our pages. I'll let you know what I get and you let me know what you get - OK?

David Goforth

WA

America Online is an Online Information Service. Cost is \$5.95 per month plus \$5 to \$10 per hour (depending on the time of day). The interface is great and very easy to use. I highly recommend it. RDEXed

Download from America Online

Making Jack Nicklaus Greatest 18 Holes of Championship Golf play faster

Requirements:

Deprotected Jack Nicklaus Golf

1 Mbyte of memory (full Apple card)

Disk copier that will copy to a RAM disk

1. Set up a 800K RAM disk inside your computer. If this is new to you, access your control panel with openapple ctrl esc. Select RAM & press return. The highlight will be on the set minimum size for RAM. Press the right arrow key until both the minimum and maximum sizes are set to 800K. Press return and exit the control panel. Turn off your machine and turn it back on. Simple, right? To reset to 0K, do the same as before but you will need to set both the min. & max. to 0 manually.

2. Run a copy program that will copy a disk to your RAM disk (I used Copy II Plus).

3. Copy the COURSE DISK of JNG to the RAM disk. What you are doing is eliminating the disk access during the game (Will be copied to Slot 5 Drive 2 which becomes the RAM disk when activated - those of you lucky dogs with 2 drives, your 2nd 3.5" drive becomes Slot 2 Drive 1).

Note: Don't turn your machine off! If you do the Course Disk you just copied to the RAM disk will be gone.

4. Now put your program disk in drive 1 and do a warm boot (openapple ctrl reset). The pro-

gram will load normally.

If you have removed the protection from the disk, it will go directly to the select skins/stroke screen. Now we're cookin'. Everything from here on will be taken from the RAM disk. This is what speeds up the game (somewhat - it's still no thoroughbred). The Power Bar and the Status Bar across the bottom of the screen will now load almost instantaneously. The graphic portion of the course (fairway view, tee shots, putting) will now load faster DEPENDING upon the amount of color changes and mixes in each line plot. Trees (T's) and shrubs (S's) still load slow but faster than before. If there aren't a lot of color changes and T's & S's the screen will load in half the time. Actual stroke play is NOT speeded up. The ball maintains its normal speed. When you finish a hole, the scorecard literally pops on the screen. Try to watch the hole numbers plot themselves on the screen - good luck. The screen containing the longest drive, # of greens, etc. also pops on the screen. One other advantage, if you want to change courses it can be done in well under 1 minute with a little practice.

Have fun with a good game of golf in less time.

Charles V. Perrien LA

Converting Print Magic Graphics to Publish It!

Requirements:

Print Magic Program and Graphics Disks

An initialized DOS 3.3 disk

2 ProDOS disks, one with ProDOS, BASIC.SYSTEM, and STARTUP

Copy II+ or similar utility for converting DOS 3.3 files to ProDOS.

A paint program for hi-res graphics such as 816/ Paint is useful but not absolutely necessary.

To Bud Myer: Here is a way to convert Print Magic pictures to use with Publish It!. I'll give a step-by-step procedure using a two drive system. If you only have one disk drive, the procedure will be a little different, but I'm sure you can follow along.

1. Boot a DOS 3.3 disk and initialize a blank disk.

You'll need it to save your Print Magic pictures.

INIT HELLO

DELETE HELLO

2. Boot the Print Magic program disk.

3. Choose keyboard from the menu. I'll use the keyboard since a lot of people don't have a mouse.

4. Insert side 2 to load the program, and press return. I'm a teacher. Although these steps may be obvious to some, I like to make things as explicit as possible!

5. Choose Page Design and press return.

6. Select Graphic and press return.

7. Choose Print Magic Graphic and press return.

8. Select Drive 2 and press return. Make sure the Print Magic Graphics disk is in Drive 2.

9. Use the openapple and downarrow key to select a graphic. I will use the graphic "Cake" for this example. Use the uparrow key to highlight "Load" and press return. You will still not see the graphic at this point.

10. Use the downarrow key to highlight "View Graphic" and press return. You should see the birthday cake on the screen.

11. Take out the Print Magic Program disk, and insert your initialized DOS 3.3 disk.

12. Press ctrl reset. This should boot the disk.

13. Save the graphic to your DOS 3.3 disk.

BSAVE CAKE, A\$4000, L\$2000

14. Copy the file "CAKE" from the DOS 3.3 disk to the ProDOS disk with the files ProDOS, BASIC.SYSTEM and STARTUP on it. At this point, you might want to edit the graphic using a paint program. I use 816/Paint for this purpose. All that needs to be done is to erase the border around the graphic and then save it to disk. 816/Paint saves high resolution graphics to page 2 (A\$4000), so it would be wise to edit the graphic now before proceeding to the next steps.

15. Boot this ProDOS disk.

16. Put the second formatted ProDOS disk in Drive 2.

BLOAD CAKE, A\$2000, L\$2000

BSAVE CAKE, A\$2000, L\$2000, S6, D2 assumes 2nd drive in slot 6, drive 2.

If you type CATALOG,S6,D1, you will see, under SUBTYPE, the starting address is A=\$4000. If you type CATALOG,S6,D2, you will see the starting address changed to A=\$2000. PUBLISH IT! will now accept this picture for loading.

If you save a number of pictures at one time, you might want to type in this short BASIC program to automate things a little at step 17. Save it on the disk with the ProDOS, BASIC.SYSTEM, and STARTUP files.


```

5 TEXT : HOME
10 DS = CHR$(4)
20 PRINT : PRINT : PRINT
30 PRINT
"ENTER THE NAME OF THE PICTURE"
40 PRINT "YOU WISH TO CONVERT FROM"
50 PRINT "PAGE 2 TO PAGE 1"
60 PRINT
70 INPUT "":FS: REM GET THE NAME
OF YOUR PICTURE YOU WISH TO
CONVERT
80 HGR : POKE - 16302,0: REM FULL
PAGE GRAPHICS PAGE 1
90 PRINT DS; "BLOAD" FS ",A$2000"
: REM LOAD THE PICTURE INTO
PAGE 1
100 PRINT :
110 PRINT DS; "BSAVE" FS
",A$2000,L$2000,S6,D2": REM
THIS SAVES YOUR PICTURE AS A
FILE IN DRIVE 2
120 TEXT : HOME
130 PRINT DS; "CAT,S6,D1": REM
THIS GETS YOU BACK TO DRIVE 1
AND SHOWS YOU THE CATALOG
140 PRINT
150 PRINT "TRY ANOTHER? (Y/N)"
160 GET AS
170 IF AS = "Y" OR AS = "y" THEN
GOTO 5
180 IF AS = "N" OR AS = "n" THEN
200
200 END
    
```

Checksums

5-\$0050	70-\$7F81	140-\$7876
10-\$64C2	80-\$FA0F	150-\$9136
20-\$FA15	90-\$9A9A	160-\$5295
30-\$D113	100-\$CB15	170-\$C572
40-\$45A8	110-\$7909	180-\$AF51
50-\$27AA	120-\$4C6E	200-\$CD05
60-\$7054	130-\$C592	

That's all there is to it! I hope I've helped some of our more inexperienced readers with this detailed procedure.

Philip C. Plunkett England

A look at self-sync bytes on Bag of Tricks

I have subscribed to COMPUTIST since 1981 and have, I believe, every issue. I regret to say however, that I have never contributed before. In the early days I felt that I could not write anything as useful or informative as the excellent technical articles and programs on Apple disk formatting that you published. Lately, now that you concentrate on specific softkeys for specific programs, I am unable to contribute as I do not have access to any recent Apple programs. The Apple II computer was never as popular in Britain as in the U.S.A. and most of the programs (particularly educational programs) that you now publish softkeys for are just not available here.

What I found most interesting about those early articles was the insight it gave into the inner working of the Apple computer. I found that cracking protected disks—slowly tracing through somebody else's obscure code—was an excellent way of learning 6502 assembly language programming. The softkeys now published do not always explain the protection mechanisms, and for those of us who do not have the particular disk the methods are of more interest than the results.

Having decided to write but not having any softkeys to offer I thought a demonstration of a certain protection mechanism, self-sync byte detection, might be an interesting topic. The actual disk used, Bag of Tricks, is old and has been superseded by an improved unprotected version.

Early nibble-copy programs were unable to determine whether disk bytes (nibbles) were written as self-sync or as normal. A whole track of disk bytes would be read into memory and then "educated guesses" were made during analysis to decide which bytes were self-sync. Certain copy protected disks would have an unexpected disk byte as self-sync and this would be checked for when the program was run. This protection scheme is no longer so effective as nibble-copy programs are now able to detect bytes that were written as self-sync. The difference between a normal (data) disk byte and a self-sync disk byte is one of timing. A normal byte of 8 bits is written to disk in 32 clock cycles (microseconds). A self-sync byte has either 9, or more usually 10 bits and is written in 36 or 40 clock cycles. The one or two extra bits used in self-sync bytes are trailing zeros. A more complete description can be found in the book "Beneath Apple DOS".

The protection schemes used on the "Bag of Tricks" disk are diverse. Briefly they are as follows:

- Not all tracks on the disk are formatted.
- The formatted tracks have 13 sectors but the address and data headers differ from normal DOS 3.2.
- The Bag of Tricks programs are not stored as files under DOS but are loaded directly, disk sectors into pages of memory, by a slightly modified DOS 3.2 RWTS routine.
- The address epilogue is written as four bytes

instead of the normal three. The last two bytes are written as self-sync bytes and are checked for during program loading. - The values of these two self-sync bytes vary from one disk to another.

- The code that checks the self-sync bytes is encrypted. It decrypts itself prior to execution and re-encrypts itself afterwards.

The protection code was found by breaking into the monitor and searching through the code. How the code worked was found by using a special tool.

When considering the attributes of an ideal tool for examining the operation of protected programs a wish list could be drawn up.

- It would be undetectable by the program running.
- It would use no memory in the computer.
- It would let the computer run at full speed at all times.
- It would provide a record of actual program flow.
- Certain pre-settable parameters would start the "recording", for example when a \$D5 is read from location \$C0EC.

The good news is that such a tool exists, the bad news is that it is expensive. It is called a logic analyzer and one such was used to examine the operation of the protection code on the Bag of Tricks disk. The logic analyzer monitors and records the status of every data, address and control line on the Apple's 6502 micro-processor. It contains a lot of memory which is used to store the status of each line for many clock cycles. The analyzer also contains a micro-computer which is used to evaluate the recorded address and data bus information and produce a disassembly listing of the actual program flow. I copied my Bag of Tricks master disk using Locksmith 4.1 which is unable to detect the self-sync bytes. I then observed how the protection code responded to the genuine master disk and to the copy.

On the disk the address field format, as observed using a track nibble reader, is as shown below.

```

D6 AA B5 FF FE AA AB AB AF FE FA DF AA F9 F9
Address D6 AA B5
Volume FF FE
Track AA AB
Sector AB AF
Checksum FE FA
Epilogue DF AA F9 F9
    
```

The last two Header bytes are self-sync.

Disassembly of self-sync byte reader

First let us look at the actual self-sync reading code (after decryption) as disassembled by the Apple II monitor. To the right I have added comments.

BF37:BD 8C C0	LDA \$C08C,X	
BF3A:10 FB	BPL \$BF37	
BF3C:C9 D6	CMP #\$D6	is this the address header?
BF3E:D0 A9	BNE \$BEE9	if not then start over.
BF40:A0 0D	LDY #\$0D	if it is then read the rest of the address marks - 13 (\$0D)
BF42:EA	NOP	
BF43:BD 8C C0	LDA \$C08C,X	more bytes to take us up to the first self-sync byte.
BF46:10 FB	BPL \$BF43	Y register acts as counter.
BF48:88	DEY	
BF49:D0 F7	BNE \$BF42	
BF4B:C9 F9	CMP #\$F9	check that the value of the third byte of the address epilogue is \$F9.
BF4D:D0 9A	BNE \$BEE9	this instruction is a time delay.
BF4F:F0 00	BEQ \$BF51	read next disk byte, if it is less than \$08 then \$F9 was self-sync.
BF51:BD 8C C0	LDA \$C08C,X	if not then start over.
BF54:C9 08	CMP #\$08	check that the fourth epilogue byte is \$F9, if not then start over.
BF56:B0 91	BCS \$BEE9	if IS \$F9 then decrement "success" counter. (Initially was set to \$04)
BF58:BD 8C C0	LDA \$C08C,X	After four good reads (\$BF50 decremented to zero) then turn motor off and return OK disk is original, not a copy.
BF5B:10 FB	BPL \$BF58	
BF5D:C9 F9	CMP #\$F9	
BF5F:D0 88	BNE \$BEE9	
BF61:CE B0 BF	DEC \$BF60	
BF64:D0 D1	BNE \$BF37	
BF66:BD 88 C0	LDA \$C088,X	
BF69:58	CLI	
BF6A:60	RTS	

The reader routine at \$BF37 looks for a \$D6 - the first byte of the address prologue. When it finds one it reads the next 13 (\$0D) bytes and then checks that the third epilogue byte is an \$F9. The actual values of the volume, track, sector and checksum are not used. After a brief time delay the data latch is examined and the value read depends upon whether the previous disk byte was self-sync or not. If it was self-sync all is well. The value of the fourth epilogue byte is then checked to see that it is also \$F9. If it is then the success counter is decremented. The success counter is initialized elsewhere to \$04 before calling the

Listing 1

Prog	Obj	Memory/	Addr / Data			
Addr	Code	Instr.	Operand	Addr / Data		
BEE9	CEB4BF	DEC	BFB4	BFB4 / 77	W	start-over location where fail counter is decremented.
BEEC	D049	BNE	BF37			
BF37	BD8CC0	LDA	C08C,X	C0EC / 1A	R	read latch but high bit not set, io byte not ready.
BF3A	10FB	BPL	BF37			
BF37	BD8CC0	LDA	C08C,X	C0EC / 6B	R	re-read latch, but byte still not clocked into latch.
BF3A	10FB	BPL	BF37			
BF37	BD8CC0	LDA	C08C,X	C0EC / D6	R	re-read latch. This time high bit is set. ie byte is ready. Is it the address prologue? YES! so do not start over. now read 13 (\$0D) more bytes.
BF3A	10FB	BPL	BF37			
BF3C	C9D6	CMP	#D6			
BF3E	D0A9	BNE	BEE9			
BF40	A00D	LDY	#0D			
BF42	EA	NOP				
BF43	BD8CC0	LDA	C08C,X	C0EC / 0A	R	
BF46	10FB	BPL	BF43			
BF43	BD8CC0	LDA	C08C,X	C0EC / 2A	R	
BF46	10FB	BPL	BF43			
BF43	BD8CC0	LDA	C08C,X	C0EC / AA	R	got second prologue byte = \$AA
BF46	10FB	BPL	BF43			
BF48	88	DEY				Y decremented to \$0C
BF49	D0F7	BNE	BF42			
BF42	EA	NOP				
BF43	BD8CC0	LDA	C08C,X	C0EC / 05	R	
BF46	10FB	BPL	BF43			
BF43	BD8CC0	LDA	C08C,X	C0EC / 16	R	
BF46	10FB	BPL	BF43			
BF43	BD8CC0	LDA	C08C,X	C0EC / 2D	R	
BF46	10FB	BPL	BF43			
BF43	BD8CC0	LDA	C08C,X	C0EC / B5	R	got third prologue byte = \$B5
BF46	10FB	BPL	BF43			
BF48	88	DEY				Y decremented to \$0B
BF49	D0F7	BNE	BF42			
BF42	EA	NOP				
BF43	BD8CC0	LDA	C08C,X	C0EC / 07	R	
BF46	10FB	BPL	BF43			
BF43	BD8CC0	LDA	C08C,X	C0EC / 1F	R	
BF46	10FB	BPL	BF43			
BF43	BD8CC0	LDA	C08C,X	C0EC / 7F	R	
BF46	10FB	BPL	BF43			
BF43	BD8CC0	LDA	C08C,X	C0EC / FF	R	got first disk byte (nibble) of disk volume = \$FF
BF46	10FB	BPL	BF43			Y decremented to \$0A
BF48	88	DEY				
BF49	D0F7	BNE	BF42			
BF42	EA	NOP				
BF43	BD8CC0	LDA	C08C,X	C0EC 0F	R	
BF46	10FB	BPL	BF43			
BF43	BD8CC0	LDA	C08C,X	C0EC 3F	R	
BF46	10FB	BPL	BF43			
BF43	BD8CC0	LDA	C08C,X	C0EC 7F	R	
BF46	10FB	BPL	BF43			
BF43	BD8CC0	LDA	C08C,X	C0EC FE	R	got second disk byte (nibble) of disk volume = \$FE
BF46	10FB	BPL	BF43			4+4 encoding used so volume = \$FF+\$FE = \$FE (254)
BF48	88	DEY				
BF49	D0F7	BNE	BF42			
BF42	EA	NOP				
BF43	BD8CC0	LDA	C08C,X	C0EC 0A	R	
BF46	10FB	BPL	BF43			
BF43	BD8CC0	LDA	C08C,X	C0EC 2A	R	
BF46	10FB	BPL	BF43			
BF43	BD8CC0	LDA	C08C,X	C0EC AA	R	got first disk byte (nibble) of track number = \$AA
BF46	10FB	BPL	BF43			Y decremented to \$08
BF48	88	DEY				
BF49	D0F7	BNE	BF42			
BF42	EA	NOP				
BF43	BD8CC0	LDA	C08C,X	C0EC 05	R	
BF46	10FB	BPL	BF43			
BF43	BD8CC0	LDA	C08C,X	C0EC 15	R	
BF46	10FB	BPL	BF43			
BF43	BD8CC0	LDA	C08C,X	C0EC 2B	R	
BF46	10FB	BPL	BF43			
BF43	BD8CC0	LDA	C08C,X	C0EC AF	R	got second disk byte (nibble) of track number = \$AF
BF46	10FB	BPL	BF43			4+4 encoding used so track = \$AA+\$AF = \$05
BF48	88	DEY				
BF49	D0F7	BNE	BF42			
BF42	EA	NOP				
BF43	BD8CC0	LDA	C08C,X	C0EC 05	R	
BF46	10FB	BPL	BF43			
BF43	BD8CC0	LDA	C08C,X	C0EC 15	R	
BF46	10FB	BPL	BF43			
BF43	BD8CC0	LDA	C08C,X	C0EC 55	R	
BF46	10FB	BPL	BF43			
BF43	BD8CC0	LDA	C08C,X	C0EC AA	R	got first disk byte (nibble) of sector number = \$AA
BF46	10FB	BPL	BF43			Y decremented to \$06
BF48	88	DEY				
BF49	D0F7	BNE	BF42			
BF42	EA	NOP				
BF43	BD8CC0	LDA	C08C,X	C0EC 0A	R	
BF46	10FB	BPL	BF43			
BF43	BD8CC0	LDA	C08C,X	C0EC 15	R	
BF46	10FB	BPL	BF43			
BF43	BD8CC0	LDA	C08C,X	C0EC 57	R	
BF46	10FB	BPL	BF43			
BF43	BD8CC0	LDA	C08C,X	C0EC AE	R	got second disk byte (nibble) of sector number = \$AE
BF46	10FB	BPL	BF43			4+4 encoding used so sector = \$AA+\$AE = \$04
BF48	88	DEY				
BF49	D0F7	BNE	BF42			
BF42	EA	NOP				
BF43	BD8CC0	LDA	C08C,X	C0EC 0F	R	
BF46	10FB	BPL	BF43			
BF43	BD8CC0	LDA	C08C,X	C0EC 3F	R	
BF46	10FB	BPL	BF43			

continued

reader routine. A failure counter is initialized to \$1000. Each time the reader "starts over" then the failure counter is decremented. If four good reads of the self-sync byte do not occur before the 4096 (\$1000) bad reads then the program decides that the disk is a copy and locks the program up with the message: "PLEASE USE ORIGINAL".

Reading a disk byte consists of two instructions, for example:

```
BF37:BD 8C C0 LDA $C08C,X read latch
BF3A:10 FB BPL $BF37 re-read if byte not ready
```

The disk hardware is serially clocking the recorded disk bits into the read latch, in the direction low bit to high bit. When a "1" bit is clocked into the high bit of the latch the byte is ready to be read. Typically the "branch and re-read" will loop three or four times before the disk byte is ready.

Now let us examine the program trace as produced by the logic analyzer when running the original disk. The listing looks rather like a disassembly listing however the program addresses are not sequential as the program when

running branches forwards and backwards. The MEMORY ADDR/DATA column shows which memory address was accessed and what the data value was. The W and R indicate whether the data was Written or Read. The listing is started just before the point where the address prologue byte (\$D6) is successfully read. I have again added comments to the listing. (See listing 1)

The critical thing that happened was that the read latch was examined a short and precise time interval after the third epilogue byte was read. Now the third epilogue byte is self-sync and has two trailing zero bits. Any byte following will clock into the read latch just those few microseconds later than if there are no trailing zero bits. By reading the latch early a low number (less than \$08) will be in the latch as the extra zero bits of the previous self-sync byte will still be in the high bits of the latch. Listing 2 is the same critical section of the program execution, but observed when the copy disk is in the drive. As the third prologue byte is not self-sync the following disk byte will be further advanced in the read latch and

is in fact now read as \$1F instead of \$07. This fails the test. (See listing 2.)

I hope this has given some insight into disk operations. When I did this "research" (a few years ago now) I had more time and went on to devise my own system of "watermarking" disks based on writing and later identifying self-sync bytes. With the nibble copiers now available the disks would be easily copyable, however at the time it was fun and I learnt something.

I must confess that my Apple II computer rarely gets switched on these days - I switched to IBM a few years ago as software I wanted to use was just not available for the Apple. Now with a 20MHz 80386 and VGA color graphics I don't miss the Apple too much. I do however miss the wealth of "inside information" that was available for the Apple - the IBM world seems to be one of users rather than experimenters and programmers.

Keep up the good work, while you continue to publish I will continue to subscribe. Please do not drop the hardware or IBM sections though, as these are now the most interesting for me.

things were explained in detail, in words of one syllable, especially things like installing special hardware to enable "Reset into the Monitor" (what is that, anyway?). I would certainly be prepared to pay around \$30 for such a package, and the people who already know such things wouldn't have to give out their hard-earned dollars for it, if it was optional. In other words, a sort of two-tier subscription - one for idiots, costing a lot, and one for whizz kids, done on the cheap!

Jim Bancroft MA

Softkey for...

Designasaurus

Designware

From BASIC CALL-151 to get into the monitor. Then type B956:00 to ignore the first byte of the address prolog. Then type B942:18 to turn off error checking. RUN COPYA and you have a normalized copy.

Softkey for...

Talking Text Writer Quest Strategy Checker Success With Typing Microzine #23 Tales of Fantasy

Scholastic

Talking Text Writer: COPYA the disk then scan for 4C 00 C6. Go back to the BD 89 C0 which precedes it at byte 9B and change it to 18 60 EA.

Quest Strategy Checker: Poke DOS to ignore the first byte of the address prolog as well as the entire address epilog, then run COPYA AS in Designasaurus described in part I of this article.

Success With Typing: COPYA the disk then do the following sector edits. Track 0 sector 3 byte \$2F change to 18 60 to ignore the data epilog. also byte \$8B same sector to ignore the address epilog. Finally track 0 sector 5 byte \$32 change to 18 60 to return without executing the gap check before the address header.

Microzine #23 and Tales of Fantasy: COPYA the disk then use the CopyII+ utilities or a sector editor to change the boot program to HELLO from whatever it currently is.

Softkey for...

Time Capsule

Learning Well (Mindscape)

The copy protection is exactly the same as the Decisions Decisions series by Tom Snyder productions.

Softkey for...

The Game Show Master Match

Computer Academic Ideas (CAI)

For new ProDOS releases of The Game Show and Master Match: COPYA the disk then search for the usual BD 8C C0 followed by CMP's to strange bytes. Each I checked used track 0 sector \$0E. There will be BEQ's BNE's and BCS's to byte \$9B. Change them to F0 00 and D0 00 and B0 00 respectively to pass the disk ID check.

Softkey for...

Biosolve

Explora-Science Whales Read-Write-Publish Explora-Classic series

D.C. Heath (Callomore or Learning Ways)

Biosolve: COPYA the disk then scan for 20 E4 21 as well as 20 36 20 and change those bytes to SEA.

Explora-Science Whales. COPYA the disk then scan for BD 89 C0 BD 8E C0 in the file called MEMORY.CONFIG. and change it to A0 00 4C F3 20 EA which is the code executed upon a successful disk check.

Read-Write-Publish and the Explora-Classic series require Bill Jetzer's ProDOS Super IOB. Fantastic job Bill. Your program reads anything that runs under modified ProDOS.

Softkey for...

Language Carnival Math Masters Garfield Companion Garfield Trivia

Sailing Through Story Problems Sailing Through Word Problems

Developmental Learning Materials

Language Carnival: COPYA the disk. In the file called STARTUP, after bloading SETUP and SETUP2, the program calls 12288 in line 5. Change the CALL to a REM and the disk check is bypassed.

Math Masters: COPYA the disk then scan for BD 89 C0 followed by C9 D5 and C9 AA and C9 BB as well as 20 C0 21 and change the BD 89 C0 to 18 60 EA.

A Loyal Reader CO

Softkey for...

Spy's Adventure in South America

Polarware

I was delighted when the softkey for Spy's Adventure in North America from "Edward Teach" worked like a charm. I also have the South America version and started trying to softkey it. I found that it had every other track protected by simply changing the address header to D4 AA 96. The even tracks had a normal header.

In Computist #52, Jason Cobb had a softkey for Ultima I with a nearly identical protection scheme. The only difference is that Spy has an altered address epilogue. Using the Ultima softkey I came up with the following 'recipe', which worked like a champ!

1. Boot your Super IOB disk.
2. Tell DOS to ignore the address epilogue.
**CALL -151
B988:18 60.
ctrl C**
3. Load Super IOB and use the Ultima controller from #52, pg. 10.

Note on '84 & '85 MECC disks

If anyone is having trouble with the '84 and '85 MECC disks using the controllers in Computist #47, try copying a normal DOS. It worked for me!! Also the softkeys for Carmen San Diego/Europe and Davidson's Spell-It worked like a charm!

Ⓢ Help! I would like to backup First Draft by Scholastic before letting my classes loose with it. Can someone help?

June Baker Germany

I am a relatively new subscriber to COMPUTIST magazine, but I have read your editorials, and the COMPUTIST Info Flyer, and would now like make a few comments and suggestions.

I think the tabloid form of the magazine is acceptable if it could be made the same size as the magazine so as to fit into normal-sized binders and/or folders. The size of the print is fine as it is, but I wouldn't like to see it much smaller, especially for the listings.

As far as the magazine content and the way it is presented are concerned, for people like myself, with no real idea of how a computer works, it is too complex and cryptic and it does not contain enough in the way of idiot-proof explanations, whereas for people still at college or school with, presumably, the time and energy to explore the inner workings of a computer, the magazine is too expensive. (I assume that most of your subscribers are students and younger people because of the continual references to the magazine costing too much.)

The magazine is described as being "For the Serious User of Personal Computers". In my opinion, I am a serious user. I have an Apple //e at home, I use a PC at work, I have used other computers at work in the past (with UNIX-based operating system, for instance). However, I am a serious user of applications programs for personal computers, in particular word processing and desktop publishing applications programs, not a programmer or a hacker. I am very interested in being able to make backup copies of my legally bought software. I am not interested in having to spend several months learning about the internal workings of the computer first. I have a living to earn, which means that my spare time is limited, and I like to spend it using my computer for the things I want to do, not in fighting to make backup copies of disks I have paid for. When I ordered your magazine and starter disk, I expected something along the lines of Copy II Plus, where you just choose which program you want to copy and the program does it for you!

I think you should change your advertisement in Nibble to make it quite clear that COMPUTIST magazine is not for the novice. Alternatively, how would it be if you prepared a special beginner's package for people like me, where

Listing 1 (continued)

BF43 BD8CC0	LDA	C08C,X	C0EC / FF	R	got first disk byte (nibble) of checksum = \$FF
BF46 10FB	BPL	BF43			Y decremented to \$04
BF48 88	DEY				
BF49 D0F7	BNE	BF42			
BF42 EA	NOP				
BF43 BD8CC0	LDA	C08C,X	C0EC 07	R	
BF46 10FB	BPL	BF43			
BF43 BD8CC0	LDA	C08C,X	C0EC / 1F	R	
BF46 10FB	BPL	BF43			
BF43 BD8CC0	LDA	C08C,X	C0EC / 3F	R	
BF46 10FB	BPL	BF43			
BF43 BD8CC0	LDA	C08C,X	C0EC / FF	R	got second disk byte (nibble) of checksum = \$FF
BF46 10FB	BPL	BF43			4+4 encoding used so
BF48 88	DEY				checksum = \$FF+\$FF = \$FF (255)
BF49 D0F7	BNE	BF42			
BF42 EA	NOP				
BF43 BD8CC0	LDA	C08C,X	C0EC / 06	R	
BF46 10FB	BPL	BF43			
BF43 BD8CC0	LDA	C08C,X	C0EC / 1B	R	
BF46 10FB	BPL	BF43			
BF43 BD8CC0	LDA	C08C,X	C0EC / 6F	R	
BF46 10FB	BPL	BF43			
BF43 BD8CC0	LDA	C08C,X	C0EC / DF	R	got first epilogue byte = \$DF
BF46 10FB	BPL	BF43			Y decremented to \$02
BF48 88	DEY				
BF49 D0F7	BNE	BF42			
BF42 EA	NOP				
BF43 BD8CC0	LDA	C08C,X	C0EC / 0A	R	
BF46 10FB	BPL	BF43			
BF43 BD8CC0	LDA	C08C,X	C0EC / 15	R	
BF46 10FB	BPL	BF43			
BF43 BD8CC0	LDA	C08C,X	C0EC / 55	R	
BF46 10FB	BPL	BF43			
BF43 BD8CC0	LDA	C08C,X	C0EC / AA	R	got second epilogue byte = \$AA
BF46 10FB	BPL	BF43			Y decremented to \$01
BF48 88	DEY				
BF49 D0F7	BNE	BF42			
BF42 EA	NOP				
BF43 BD8CC0	LDA	C08C,X	C0EC / 0F	R	
BF46 10FB	BPL	BF43			
BF43 BD8CC0	LDA	C08C,X	C0EC / 3E	R	
BF46 10FB	BPL	BF43			
BF43 BD8CC0	LDA	C08C,X	C0EC / 7C	R	
BF46 10FB	BPL	BF43			
BF43 BD8CC0	LDA	C08C,X	C0EC / F9	R	got third prologue byte = \$F9
BF46 10FB	BPL	BF43			CRITICAL PART COMING UP NOW!
BF48 88	DEY				Y decremented to \$00
BF49 D0F7	BNE	BF42			Y = zero so don't branch.
BF4B C9F9	CMP	#F9			is third prologue \$F9?
BF4D D09A	BNE	BEE9			YES! so do not start over.
BF4F F000	BEQ	BF51			short time delay.
BF51 BD8CC0	LDA	C08C,X	C0EC / 07	R	read latch NOW and get \$07.
BF54 C908	CMP	#08			is it less than \$08?
BF56 B091	BCS	BEE9			YES! so do not start over.
BF58 BD8CC0	LDA	C08C,X	C0EC / 1F	R	
BF5B 10FB	BPL	BF58			
BF58 BD8CC0	LDA	C08C,X	C0EC / 7C	R	continue to clock in last epilogue byte.
BF5B 10FB	BPL	BF58			which is \$F9 as expected.
BF58 BD8CC0	LDA	C08C,X	C0EC / F9	R	
BF5B 10FB	BPL	BF58			
BF5D C9F9	CMP	#F9			
BF5F D088	BNE	BEE9			
BF61 CEB0BF	DEC	BFB0	BFB0 / 03	W	all OK so decrement success counter.

Listing 2

Prog	Obj.	Memory/			
Addr	Code	Instr.	Operand	Addr / Data	
BF43 BD8CC0	LDA	C08C,X	C0EC / F9	R	got third prologue byte = \$F9
BF46 10FB	BPL	BF43			
BF48 88	DEY				CRITICAL PART COMING UP NOW!
BF49 D0F7	BNE	BF42			Y = zero now.
BF4B C9F9	CMP	#F9			is third prologue \$F9?
BF4D D09A	BNE	BEE9			YES! so do not start over.
BF4F F000	BEQ	BF51			short time delay.
BF51 BD8CC0	LDA	C08C,X	C0EC / 1F	R	read latch NOW and get \$1F
BF54 C908	CMP	#08			is it less than \$08?
BF56 B091	BCS	BEE9			NO! \$F9 was not self sync.
BEE9 CEB4BF	DEC	BFB4	BFB4 / C3	W	decrement fail counter
BEEC D049	BNE	BF37			if not zero try again.

Garfield Companion: Both sides contain exactly the same programs. One side will COPYA normally. The other will need COPYA poked to ignore epilogs. Once copied scan the file L5 for BD 89 C0 which leads into the disk check. Change it to 18 60 EA.

Sailing Through Story Problems and Sailing Through Word Problems (written by NeoSoft): Both programs will COPYA if poked to ignore epilogs. They both use the SSPROTSS programs as their protection scheme. Use the CopyII+ utilities or a sector editor to change the boot program to DLMPirate1 and the disk check will be bypassed. If the disk will not run, check track \$11 sector 0 byte 1. If it is not \$11 change it and write the sector back. I didn't see this on the disks I did, but I know it is standard procedure in SSPROTSS protection schemes.

Garfield Trivia: COPYA the disk and scan for BD 8E C0 BD 89 C0 and change the BD 89 C0 to 18 60 EA. For your information, each data disk of categories created has a file created to store the disk name. The name stored consists of the ASCII code of the characters typed minus 31 (\$1F). If you want to edit the original data disk, you must change at least one byte of its disk name. After editing be sure to change the name bytes back to normal or the disk will not run as the program's default categories. PS: the original data disk's name is "Hmenlimhi" which decodes as "Infomania", a rather descriptive statement of copy protection schemes.

Softkey for...
**WISCR Intelligence Test
Mathematics Today**

Harcourt Brace Jovanovich (HBJ publishing)

Mathematics Today: This program requires use of the swap controller for tracks 3 to 35. The disk you initialize must use Diversi-Dos, since the programs make calls into that DOS in order to operate. Use a sector editor on your copy. Edit track \$11 sector 0 (the VTOC of the disk) at byte 1 change the 0 to \$11 this will make the catalog readable. Inside the program HELLO, look for a poke to 40XXX,0 where XXX can be various numbers. These pokes will strategically place BRK's within normal Diversi-DOS but are in uncalled locations within HBJ's modified version of that DOS.

WISCR Intelligence Test: This disk uses EA D3 9C instead of D5 AA 96 as its address prolog and uses various two byte sets as its data epilogs. Run COPYA. Hit control-reset. Type 70. Type CALL-151 to get into the monitor and poke DOS as necessary to change the address prolog read bytes and to ignore the data read epilogs. Type 3DOG and hit return to get back into basic. Type RUN to COPYA the disk. The copy will need editing before it can run. The file WP.FUNC contains C9 EA change it to C9 D5 also change C9 D3 to C9 AA and change C9 9C to C9 96. This was a check to see what is being used as the address prolog on the disk. The file WISC in line 10445 opens a file called COUNT which reads a value, lowers it by one and, rewrites the file. This is the count of allowed uses of the disk. It may be 10 or 100 depending on your license. There is also a GOSUB within the WISC file to a routine at 59000. This subroutine checks the WP.FUNC file to see if you have substituted D5 for EA in the address prolog. This is the third checks for the altered prolog. Change the first working line of this subroutine to RETURN and the check will not be done. Use copy II+ Utilities to copy a normal DOS onto the copy. (it is rather fussy as to what DOS it will run under.) FINALLY you have a working copy on a normal disk which you can archive as needed with your student's records (one student to a disk) if you prefer.

Softkey for...
**Study for Success
Test Taking Made Easy
Building memory Skills**

MicroComputer Educational (MCE):

COPYA the disks. The copies all require the same edits to track 0, sector 5 in order to run. At byte \$AB put EA A9 D7. At byte \$B6 put EA A9 F9. These edits will load the accumulator with the proper values of a successful disk check. At byte \$C2 put EA EA EA. This will bypass the disk read so that the values you just loaded to the accumulator get stored into memory at \$70 and \$71 for later use at \$3F0 and \$3F1. This will break it out of the endless loop from \$BB34 to \$BB38 in which even the best bit-copies usually hang.

Softkey for...
**Houghton Mifflin Math
Courseware**

Houghton Mifflin

COPYA the disks. Check Track \$11 sector 0 byte 1 to be sure that the value is \$11. If it is not, change it to \$11. This entire series uses the SSPROTSS protection as described in a previous issue for Walt Disney Comic Strip Maker. However in this program the HELLO file can not be listed or captured by the usual method. After the SSPROT files have done the disk check and HELLO load and decode, interrupt the program.

Use the Fingerprint+ interface card or the Senior Prom on my IIe for that purpose. Type CALL-151 to get to the monitor. Type 27FE < 7FE.13FFM and hit return. This will move HELLO to a safe location. Then insert a bootable slave disk with no hello program and type C600G to boot it. Get back into the monitor if necessary (CALL-151), type 7FE < 27FE.33FFM and hit return. This will bring the decoded HELLO back from safe keeping. Type AF: 4D 11 to set the proper end of basic file pointer and hit return. Now SAVE HELLO and you have a working copy.

Softkey for...
**McGraw Hill Compucat
Quizware**

McGraw Hill

Poke COPYA to ignore epilogs and copy the disk. The disk does a startup to a VTOC and catalog on track 7. The file DEMO COMPUCAT in line 250 pokes the VTOC location to 35 and the epilog to 170 change the 35 to 17 and eliminate the epilog poke. Since the VTOC exists on track 17 already, it will not be necessary to copy it from track 35. SAVE DEMO COMPUCAT and you have a working disk. Be sure to load and save DEMO COMPUCAT using their DOS since it is stored by the track 7 catalog.

Softkey for...
Picture Perfect

Mindplay

Locksmith Quick Copy will copy this disk ignoring and clearing the nibble counted track. Then in "FILE#1" on track 6 sector 1 I found BD 8C C0 10 FB C9 AD F0 08. Change the F0 to D0. Later in the same sector is 8D 20 C9 20 D0 0B. Change the D0 to F0. Then in the "ZURP" file is a BD 8C C0 at bytes 58 to 5A and later a 4C 3F 93. Change all those bytes to EA. Also change all D0 XX and F0 XX, where XX can be any value, to D0 00 and F0 00 respectively within that section of code.

Softkey for...
Transylvania

Penguin-Polarware

COPYA both sides of the disk. Then on side 2 you will find 4C FB 1A which is a jump onto itself giving an endless hang. Change the code to EA EA EA.

Softkey for...
Delta Drawing 3.33

Spinnaker

This version is in ProDOS look for 6E A8 7C and replace it with 4C 77 6F. It was in ProDOS block \$21 on the disk I checked. The code in question builds disk check code in memory as it decodes. Upon passing the disk check, the program then jumps to 6F77 (information courtesy of the Senior Prom.) If you own a IIe and don't have the PROM, get it!

A Bug in Prentice Hall Science Courseware

First Prentice Hall Science Courseware described by me in Issue #61. I pointed out which code to search for on track 5, sector 3. You still look for the same code to find the protection routine but then locate A0 00 DD 88 C0 60 in the sector and change it to A0 01 EA EA EA 60. This method worked on all 12 programs in the set which I tried. My original edits worked on only 3.

An accelerated IIe & EDD 4

Also in Issue #61 Brian Troha gives a fix for his EDD4 softkey to allow it to work on a GS. That same fix also allows it to run on an accelerated IIe.

**An accelerated IIe & Carmen SanDiego
(World)**

Armed with Brian's observations I checked the CAR file created in the Computist softkey of Where in the World is Carmen SanDiego. If you locate the sector containing 38 AD 52 C0 58 and 4C 03 60, you will notice a long string of EA's followed by a D0. Extend the string by adding EA EA to remove the BNE (\$D0) and Carmen works fast and furious on the GS and accelerated IIe's. It no longer hangs during the boot.

A note on Stickybear encrypted sector

After doing numerous successful Stickybear softkeys by using Issues 26 & 52's directions, I noticed something. The sector we decode and write back to the disk, once decoded, is an exact duplicate of another sector on the disk which exists originally in normal unencrypted form. If we knew the sector in question we could save the trouble of capturing and rewriting the encrypted code in normal format. Can anyone determine a pattern to the memory page and corresponding unencrypted sector?

Finding the licensee's name in GEOS

Finally GEOS. I noticed that you can quickly find the licensee's name on the working disks by scanning for a string of \$26's. On the boot disks which are encrypted check ProDOS block \$10.

The end of the name will have a string of &&& in ASCII. The letters of the name are 2 previous in the alphabet to what was typed at the keyboard.

A Personal Observation

If anyone thinks Computist is overpriced at \$32.00 a year they are definitely not using the magazine to its full power. Computist saves me days or weeks, not hours, of work during an already too busy work year.

If we all contribute we can save each other months of tedious code searches and substitution trials.

Maverick Game Hacker II

To Jan Recourt (issue #70) on Moebius: I believe I can answer your question correctly. Often times, good monks get killed or kill themselves. They have drowned or been slaughtered from walking off to far. I believe that is your problem. Most likely you are leaving the door to their prison open and they wander off into the water and kill themselves. Also, maybe it was a mistake, but you said "MONK", not "MONKS". There should be two monks. Close the door to their prison (if you did indeed leave it open, if not, either someone or something is killing them, or I'm wrong) and go to the palace and wait. I'm sorry if this doesn't work, but I've won it countless times and this solution is the only one that I can think of.

As for Tower of Myraglen: It has been a long time since I have played it and I lost my disk somewhere. I'm not sure what your problem is, but if 6 is the level I'm thinking of, I think I know. You should, if I'm correct, have an Amulet of Defense in your possession. Try starting level 6 over, and this time do not get the Amulet. Sorry if this doesn't work. Remember that when they say defense they mean that the Tower is not yours. I fell for this trick also, and as a result, had to start from level one because I had saved it with the Amulet. Also, if you do not have a book on the ideas of the creator of the Tower, you better get one. It's found in a cabinet along one of the room's walls.

Dave Grenda LA

Where can I get a Computist electronic index? I'm looking for an index that lists cracks, playing tips, APTs, etc., and has periodic updates (once or twice per year) at a nominal cost.

There's a free Appleworks database on the Computist BBS and on the library disks (starting with this issue). It is kept up to date by J.L. Walters. There is also an excellent Appleworks Database, with updates, for sale by David R. Hopkins. See the classifieds in this issue. RDEXed

How can I put DOS 3.3 programs onto a ProDOS disk? I have several binary DOS 3.3 programs on 5-1/4" disks (the programs don't access the disk after loading and aren't copy protected). The programs run fine from a DOS 3.3 formatted disk but I have problems with some of them on a ProDOS disk. I get a NOBUFFERS AVAILABLE message when I BLOAD/BRUN some of the larger programs from the ProDOS disk. I have a IIgs with an Applied Engineering 1.5 meg GS RAM card, so I know the machine isn't memory limited. What can I do to get these programs loaded and running from a ProDOS disk?

That's two questions so let's look at them in order. Putting DOS 3.3 programs onto a ProDOS formatted disk is easy, just use the convert program supplied on your ProDOS system disk. Getting those DOS 3.3 programs to work under ProDOS is not easy. DOS 3.3 is an operating system, not a disk. When we say "DOS 3.3 disk", we mean a disk that was formatted by the Disk Operating System version 3.3 (DOS 3.3). When you boot a DOS 3.3 disk, you are loading the DOS 3.3 Operating System Software. DOS 3.3 and ProDOS are two very different Disk Operating Systems. A program written for DOS 3.3 will have different file commands and may reside in portions of memory or do peeks, pokes or calls into memory that is used in other ways by ProDOS. Some BASIC programs can be converted if they don't do direct calls into DOS 3.3, but anything along that line will have to be done on a program by program basis. There are no blanket answers. RDEXed

How can I get Skyfox to run on my IIgs? Skyfox runs fine on my old II+, but it won't even boot properly on my new IIgs. Is there anything I can do to make it run on my IIgs or am I just S.O.L.?

Are there any Apple users groups in the Shreveport area? If there are any Apple users groups in my area please contact me at: 3100 Fairfield Ave - #5B Shreveport, LA 71104

Comments

Publish an annual Computist for archival purposes. I don't like the tabloid format because it can't be used as a long term reference. One solution, which has been suggested before, is

publishing a periodic (annual?) "book" containing everything from previous "tabloid" issues. Charge subscribers only what it costs plus a small mark-up, they've already paid once for the info.

I like the idea of the BBS as long as it doesn't replace the magazine. BBS info won't reach all subscribers because either they don't have a modem or can't access the board enough. Please make sure BBS info eventually makes it to print!

Determine if an "800" number is feasible for the BBS. I don't know the cost of a toll-free number, but I'd pay a "reasonable" fee for it. There may be enough subscribers (those not living in the Northwest) who would pay an annual fee to save on long distance phone bills. I don't know if a dual line is possible/practical, but the BBS could have a second line for others who don't want to pay the annual fee.

An "800" line isn't really free, it just means that someone else is paying for the call. RDEXed

David R. Hopkins

Missing Issues?

I would like to clarify a comment made by Mr. Toshiakazu Yamamoto, printed in issue #70, page 16 (third column/fourth paragraph), regarding the Computist Super Index. He stated that the Computist Super Index increased the value of his collection of Computist issues and that "...several early versions are missing though..." The latter statement could be construed as meaning the Computist Super Index is missing issues. I want to assure everyone that Mr. Yamamoto's "collection" is what is missing issues. The Computist Super Index covers all issues of Computist, including the original "Core" and "Hardcore" issues and updates.

Mr. Yamamoto continues by stating the Computist Super Index missed Tim Stahmer's (article?) requesting the name of the publisher of "Disk Muncher" in issue #22, page four, fourth paragraph. Except for Mr. Yamamoto's biting comment, "And the second mystery is 'Complete' Super Index looks to have missed listing this interesting article," this is an excellent recommendation. Since issue #60, I have been itemizing each comment and question as I add each issue to the Computist Super Index. I am reviewing all previous issues (in reverse order) and itemizing as needed. More on this and the Computist Super Index later. Thank you for your suggestions and generous support!

Zorro FL

Put Autoduel on a 3.5" diskette

Requirements:

Apple IIe, IIc or IIgs
Deprotected copy of Autoduel (COMPUTIST #36)
3.5" disk drive (Apple UniDisk 3.5" or Central Point 3.5" drive, etc.)
Copy II Plus v7-up (both 3.5" and 5.25" versions)
UniDOS Plus by Microsparc, Inc.
Blank 3.5" microdisk

I'm sure most COMPUTISTs are familiar with Autoduel, Origin Systems' excellent fantasy role-playing game. Those who have the game know that it is easily crackable with the softkey by Charles Taylor in COMPUTIST #36. Several factors that make Autoduel easy to normalize also make it possible for transfer to a 3.5" microdisk for convenience. One, the deprotected version has a file structure identical to that of DOS 3.3. Two, because of this normal file structure, the program loads itself by utilizing a DOS rather than accessing the disk directly with separate machine language routines. Because of this similarity in operation with DOS 3.3, moving Autoduel to a microdisk is quite possible, and not too difficult.

Stage One - From Questions to Answers

To begin, I booted up my deprotected copy of Autoduel and pressed Ctrl-Reset after the drive stopped. The boot file B(ctrl)A)T began at \$800 in memory, so I started examining the code there. After loading the slick title routine, the program waits for a keypress and then loads more code which asks you how many 5.25" disk drives are in your system. Obviously, to amalgamate the entire program into one microdisk, the number of drives is irrelevant, and so the first step is to modify Autoduel so that it will not ask for this unnecessary information, and then eliminating the 'Insert disk...' messages will follow.

Autoduel prints characters on the screen using a unique HRCG (Hi-Res Character Generator). A message to be displayed on the screen is preceded by a JMP to the HRCG which then reads the string of text following the jump. Several of these messages existed from the first screen after the title on into the rest of the game. The ones to eliminate were the first two, which ask for the number of drives and tells you which side to insert into which drive. Although I'm not a hard-core machine language programmer, I found both messages and they appeared as follows:

First Message - "Will you be using 1 or 2 drives?"


```

08CC:20 3C 8C JSR $8C3C goto print routine,
message follows
08CF:8D D7 E9 STA $E9D7 <ret>
08D2:EC EC A0 CPX $A0EC II
08D5:F9 EF F5 SBC $F5EF, Y you
08D8:A0 E2 LDY #$E2 b
08DA:E5 A0 SBC $A0 e
08DC:F5 F3 SBC $F3,X us
08DE:E9 EE SBC #$EE in
08E0:E7 ??? g
08E1:A0 B1 LDY #$B1 1
08E3:A0 EF LDY #$EF o
08E5:F2 ??? r
08E6:A0 B2 LDY #$B2 2
08E8:A0 E4 LDY #$E4 d
08EA:F2 ??? r
08EB:E9 F6 SBC #$F6 iv
08ED:E5 F3 SBC $F3 es
08EF:BF ??? ?
08F0:8D 8D 00 STA $008D $00 at $8F3 signifies
end of message

```

Second Message - "Insert disk B into drive X and press a key." (X either 1 or 2)

```

090A:20 54 8C JSR $8C54
090D:20 3C 8C JSR $8C3C goto print routine again
0910:8D C9 EE STA $EEC9 message begins here
0913:F3 ???
0914:E5 F2 SBC $F2
0916:F4 ???
0917:A0 E4 LDY #$E4
0919:E9 F3 SBC #$F3
091B:EB ???
091C:A0 C2 LDY #$C2
091E:A0 E9 LDY #$E9
0920:EE F4 EF INC $EFF4
0923:A0 E4 LDY #$E4
0925:F2 ???
0926:E9 F6 SBC #$F6
0928:E5 A0 SBC $A0
092A:B2 ??? varies; may be either
092B:A0 E1 LDY #$E1 $B1 or $B2 depending
092D:EE E4 A0 INC $A0E4 on # of drives.
0930:F0 F2 BEQ $0924
0932:E5 F3 SBC $F3
0934:F3 ???
0935:8D E1 A0 STA $A0E1
0938:EB ???
0939:E5 F9 SBC $F9
093B:AE 8D 00 LDX $008D message ends here at
the $00
093E:20 FD 09 JSR $09FD jump to routine which
awaits keypress
0941:A9 02 LDA #$02 # of drives, 2 in this
case
0943:8D 67 0F STA $0F67 store for use elsewhere
0946:A9 C2 LDA #$C2 side letter, B in this case
0948:8D 82 0F STA $0F82 store it
094B:85 F8 STA $F8
094D:20 5D 0F JSR $0F5D go check if side B in
drive
0950:90 03 BCC $0955 side B in, execute rest of
program
0952:4C 0A 09 JMP $090A side B not in, so print
message again
0955:20 A2 0D JSR $0DA2 continue execution

```

Notice from these listings the JMP to the print routine just before the string of ASCII characters, and then the BRK opcode (00) at the end of each message to terminate any further printing to the screen of the code following the text. To decipher these messages, I used Beagle Bros' Peeks, Pokes, and Pointers chart, which was invaluable in the process.

In the first listing, *Autoduel* asks how many drives are in your system. Pay close attention to the second listing; in this case, the program jumps to SF5D to check if side B is present, and if not, it JMPs back to \$90A to print the message again and wait until disk B is in drive 1. If side B is present, however, the program continues executing. Therefore, we must modify this section so that it disregards this message entirely and simply continues execution at \$DA2, while depositing a \$01 at \$F67 and a \$C1 at \$F82 and \$F8 to be sure that any references to side B and/or the number of drives is eliminated. Boot Copy II Plus, and modify your deprotected copy of *Autoduel* with the Sector Editor:

Trk	Sci	Byte(s)	From	To
\$17	\$0F	\$46	02	01
		\$4B	C2	C1
		\$56-58	4C 0A 09	EA EA EA

The number of drives in your system is now unimportant, so change the following to bypass this unnecessary query:

Trk	Sci	Byte(s)	From	To
\$16	\$00	\$CD-CF	20 54 8C	EA EA EA
		\$D0-D3	20 3C 8C	4C 41 09

Here is a section of code which is of particular importance:

```

9A20:A5 E5 LDA $E5 load side # into
accumulator
9A22:C9 B2 CMP #$B2 was it side 'B'?
9A24:F0 37 BEQ $9A5D yes, side B in drive, so
branch
9A26:A9 01 LDA #$01 side B not in drive, so
print message
9A28:8D 73 9A STA $9A73
9A2B:20 52 E0 JSR $E052
9A2E:20 19 97 JSR $9719 go to message print
routine
9A31:8D C9 EE STA $EEC9 message begins here
9A34:F3 ??? message reads "Insert
disk X and press

```

```

9A35:E5 F2 SBC $F2 a key." (X is either A or
B)
9A37:F4 ???
9A38:A0 F3 LDY #$F3
9A3A:E9 E4 SBC #$E4
9A3C:E5 A0 SBC $A0
9A3E:C2 ???
9A3F:A0 E1 LDY #$E1
9A41:EE E4 A0 INC $A0E4
9A44:F0 F2 BEQ $9A38
9A46:E5 F3 SBC $F3
9A48:F3 ???
9A49:A0 E1 LDY #$E1
9A4B:A0 EB LDY #$EB
9A4D:E5 F9 SBC $F9
9A4F:AE 8D 8D LDX $8D8D
9A52:00 BRK end of message

```

You will discover that this is the message preceding and following any arena combat if you have specified a one-drive system. After loading the contents of location \$E5 into the accumulator, *Autoduel* checks to see if side B is in drive 1, and if so it then continues execution of the program by branching to \$9A5D. If side B is not in the drive, however, the program displays the string beginning at \$9A31 and awaits a keypress which then initiates another search for the proper disk side. To bypass this portion of the code, we must tell *Autoduel* that side B is already in drive one (combined with side A on the microdisk), and have it continue execution from there. This is accomplished by replacing the JSR \$9719 with a JMP \$9A5D, so make these changes:

Trk	Sci	Byte(s)	From	To
\$1B	\$0D	\$32-34	20 19 97	4C 5D 9A

If you boot up the *Autoduel* disk with these modifications, it should skip asking questions entirely and immediately attempt to load files from what it thinks is side B.

Stage Two - Fixing Filenames

As you have already noticed, I have selected UniDOS Plus as the operating system for *Autoduel*, not ProDOS. Unfortunately, *Autoduel* seems to be incompatible with ProDOS, in that it messes up upon booting. I'm optimistic that someone will bridge this incompatibility gap, so I have provided the following modifications to *Autoduel's* filenames. If compatibility with ProDOS is achieved, these steps will save a good amount of time for the when conversion becomes possible.

Unlike DOS 3.3, ProDOS does not allow control characters (i.e. Ctrl-A, Ctrl-Z, etc.) in its filenames. First off, boot up Copy II Plus (5.25" version), and from the utilities menu, select the CATALOG W/HIDDEN CHARACTERS option. Place side A of *Autoduel* into drive 1 and catalog the disk. Notice that the second character in each of the files is an inverse 'A' (Ctrl-A), which would remain unseen if the disk were CATALOGed normally. Next, go to the RENAME FILES option and rename all files as they appear in the listing. Although there seem to be no control characters in this normal catalog, they nevertheless do exist, and all files on side A have a Ctrl-A in their filenames. For example, the file which in actuality is named A(Ctrl-A)1, would be renamed A1, without its control character. Repeat this renaming procedure for each file.

If *Autoduel* is now booted, you would not expect it to run at all since each of the filenames on side A have been changed from what the program expects them to be. Fortunately, *Autoduel* uses that same character generating routine to output DOS commands, such as BLOAD and BRUN. Here is an example of such usage:

```

0DA2:20 30 8C JSR $8C30 jump to print routine
0DA5:8D 84 C2 STA $C284 DOS command reads
"BLOAD CR,$8A00"
0DA8:CC CF C1 CPY $C1CF
0DAB:C4 A0 CPY $A0
0DAD:C3 ???
0DAE:D2 ???
0DAF:AC C1 A4 LDY $A4C1
0DB2:B8 CLV
0DB3:C1 B0 CMP ($B0,X)
0DB5:80 8D BCS $0D44
0DB7:00 end of command

```

At the beginning of the message, note the \$84 present at \$DA6; this is a Ctrl-D, and it precedes all DOS commands. Whenever accessing the disk, *Autoduel* uses this basic format for the loading and/or saving of files. To allow *Autoduel* access to the newly modified files, we must change the filenames within the commands so that they do not reflect the control characters which were once part of them. To do this perform the following sector edits:

Trk	Sci	Byte(s)	From	To
\$0C	\$02	\$C3-C5	CD 81 C4	A0 CD C4
	\$04	\$1B-1D	C1 81 CD	A0 C1 CD
		\$33-35	C1 81 C4	A0 C1 C4
\$12	\$0C	\$0D-0F	D3 81 D0	A0 D3 D0
	\$0D	\$AD-AF	D0 81 CC	A0 D0 CC
		\$BF-C1	D0 81 CD	A0 D0 CD
\$16	\$00	\$28-2A	CE 81 D3	A0 CE D3
		\$3A-3C	D3 81 B1	A0 D3 B1
		\$4C-4E	CD 81 C4	A0 CD C4
		\$5E-60	D3 81 B2	A0 D3 B2
		\$70-72	D4 81 C2	A0 D4 C2
		\$87-89	CD 81 D3	A0 CD D3

By scanning the disk for any DOS commands, such as BLOAD and BSAVE, I found these bytes. Notice the \$81 in each entry under

the 'From' column; this is the Ctrl-A which was formerly part of the filenames, and it is eliminated by moving the first byte of the filename so that it will replace the Ctrl-A, and then placing an extra space (\$A0) at the beginning of each, which does nothing but fill the gap.

If you have CATALOGed side B, you noticed that two filenames are identical on each side. There is a file called 'A1' and one called 'A2' on each side of the *Autoduel* disk. If both sides are to be combined into one, there can be no identical files, so boot Copy II Plus, go to the RENAME FILES option, and rename files 'A1' and 'A2' on side B to 'A4' and 'A5' respectively. Then, use the sector editor to modify these bytes:

Trk	Sci	Byte(s)	From	To
\$0D	\$05	\$16-17	C1 B2	C1 B5
\$1B	\$09	\$F9-FA	C1 B1	C1 B4

Using the SCANTEXT option on the bitcopy menu of Copy II Plus, I searched side B for the filenames in question, 'A1' and 'A2'. If they were preceded by BLOAD or BRUN command, then changing the filenames to 'A4' and 'A5' would prove effective, and the proper files would be run.

By now, all problems of combining the two disk sides are eliminated, so the actual transfer from floppy to microdisk can begin.

Stage Three - The Final Fix

All that is left is to transfer the files from sides A and B of *Autoduel* to one microdisk. Boot up UniDOS Plus, and after the DOS is loaded, place a blank 3.5" disk into the drive and type the following:

```

NEW
POKE 44452,255:POKE 44513,67 modify DOS to
hide all unlocked files
POKE 40514,52 tells DOS to run binary file when
booted
INIT Autoduel
DELETE Autoduel

```

Boot Copy II Plus (3.5" version), and proceed to copy all files from sides A and B to the microdisk. After this copying process is complete, RENAME the file BT as *Autoduel*, LOCK *Autoduel*, and voila, a complete and working copy of *Autoduel* on a 3.5" disk!

The Cookbook Method

1. Make a copy of the *Autoduel* disk using the softkey in COMPUTIST #36.
2. Boot Copy II Plus, and make some changes to the side A of the deprotected copy of *Autoduel* with the sector editor:

Trk	Sci	Byte(s)	From	To
\$0C	\$02	\$C3-C5	CD 81 C4	A0 CD C4
	\$04	\$1B-1D	C1 81 CD	A0 C1 CD
		\$33-35	C1 81 C4	A0 C1 C4
\$12	\$0C	\$0D-0F	D3 81 D0	A0 D3 D0
	\$0D	\$AD-AF	D0 81 CC	A0 D0 CC
		\$BF-C1	D0 81 CD	A0 D0 CD
\$16	\$00	\$28-2A	CE 81 D3	A0 CE D3
		\$3A-3C	D3 81 B1	A0 D3 B1
		\$4C-4E	CD 81 C4	A0 CD C4
		\$5E-60	D3 81 B2	A0 D3 B2
		\$70-72	D4 81 C2	A0 D4 C2
		\$87-89	CD 81 D3	A0 CD D3
		\$CD-CF	20 54 8C	EA EA EA
		\$D0-D3	20 3C 8C	4C 41 09
\$17	\$0F	\$46	02	01
		\$4B	C2	C1
		\$56-58	4C 0A 09	EA EA EA
\$1B	\$0D	\$32-34	20 19 97	4C 5D 9A

3. Make these changes to side B:

Trk	Sci	Byte(s)	From	To
\$0D	\$05	\$16-17	C1 B2	C1 B5
\$1B	\$09	\$F9-FA	C1 B1	C1 B4

4. Go to the RENAME FILES option on the Copy II Plus Utilities Menu, and place side A of *Autoduel* into the drive. Rename all files as they appear on this listing. In other words, rename file BT as BT, PL as PL, etc. for all side A files. This eliminates the Ctrl-A characters in the filenames.

5. On side B, rename the file A1 to A4 and the file A2 to A5.

6. Boot up UniDOS Plus, and after the drive has stopped, insert a blank, unformatted 3.5" disk into the drive and type the following:

```

NEW
POKE 44452,255:POKE 44513,67
POKE 40514,52
INIT Autoduel
DELETE Autoduel

```

7. Copy all files on sides A and B of the deprotected *Autoduel* 5.25" disk to the formatted 3.5" disk with Copy II Plus.

8. Go to RENAME FILES on the Utilities Menu, and rename the file BT on the 3.5" disk to *Autoduel*.

9. Go to LOCK/UNLOCK FILES on the Utilities Menu and LOCK *Autoduel*.

That's it.

Closing Notes

This enhancement to *Autoduel* should perform as stated. Unfortunately, I have none of the 3.5" implements I listed in the requirements, and so I haven't been able to test these steps firsthand. WHAT?! (you gasp). Nope, I don't own a 3.5" drive (yet), nor do I own UniDOS Plus. However, if UniDOS Plus is as close to DOS 3.3 as the ads in Nibble magazine state, then you

should be successful. Unfortunately, with this success may come the torture of painfully slow disk access from UniDOS, since it is possible that Microsparc may have kept this annoyance along with the more productive features of DOS 3.3. To remedy this if such is the case, I strongly recommend that whomever encounters this problem should try to merge UniDOS Plus with ProntoDOS, perhaps by swapping the sector interleaving patterns and then formatting the blank 3.5" disk.

Also, if someone does eventually make *Autoduel* compatible with ProDOS, I'd like to hear from you - I've already written a program to load *Autoduel* from a 3.5" disk to RAM in ProDOS, which would work really well if only *Autoduel* were ProDOS compatible.

Advanced Playing Technique for...

Autoduel

Origin Systems, Inc.

If you annoy being met with the 'It's closed...' message and hate having to waste time sleeping over at the Truck Stop before visiting the various shops and offices in the cities, change the following bytes with Copy II Plus' sector editor on side B of the *Autoduel* disk:

Trk	Sci	Byte(s)	From	To
\$1B	\$09	\$83-85	20 54 8C	EA EA EA
		\$8E-90	20 3C 8C	4C B7 0B

Now all facilities in the *Autoduel* towns are open 24 hours a day, seven days a week (just like 7-11). Of course, you still must stay at the Truck Stop occasionally to await duels in the Arena.

Softkey for...

Silicon Dreams

Rainbird

Requirements:

Copy II Plus v7-up
Super IOB w/SWAP.CON
NMI (Non-Maskable Interrupt) card
Blank, formatted DOS 3.3 diskette

Silicon Dreams is a fairly interesting adventure in which you, as your alter-ego Kim Kimberly, partake in three related adventures revolving around a planet called Eden, which is colonized and prepared for human habitation by a force of intelligent robots. In the first adventure, *Snowball*, you have been awakened from hibernation aboard the colonization ship called *Snowball 9*, and must find a way to divert the ship from an imminent collision with Eden. In *Return to Eden*, the second adventure, you have saved the *Snowball* from destruction, and now you must clear yourself of a crime which you didn't commit. In the final adventure, *Worm in Paradise*, you are a citizen of the Megopolis of Enoch on the planet Eden, and must discover the mysteries below the tranquil city above.

The Protection

Upon examination of the protection with *Trax* (Bag of Tricks), altered checksums were revealed along with non-sequential sector numbers, but, other than these there were no other copy-protection schemes. Upon capturing the foreign RWTS, I discovered it was very similar to DOS 3.3, so I decided that using it in conjunction with *Super IOB* was the best way to go.

The Crack

1. Boot the *Silicon Dreams* diskette, make a menu selection, and wait for the disk drive to stop.
2. Break into the monitor with your NMI card, modify the resident RWTS, and move it to a safe location.

```

ctrl reset
B98A:00 Ignore address checksum
B92E:00 Ignore data checksum
B942:18 Ignore read errors
1900<B800.BFFFM Move RWTS to $1900

```

3. Boot up a DOS 3.3 disk and save the RWTS.

```

6 ctrl P
BSAVE RWTS.SILICON,AS1900,LS800

```

```

LOAD SUPER IOB
EXEC SWAP.CON
10010 PRINT CHR$(4) "BLOADRWTS.SILICON,
AS1900"

```

5. Insert a blank disk into drive 2 and run *Super IOB*.

```

RUN

```

6. Boot *Copy II Plus* and from the main menu select the 5.25" BITCOPY option.

7. Select the MANUAL SECTOR COPY option from the bitcopy menu.

8. Copy tracks \$00 to \$02 from the original to the copy so that the program will boot as it usually does.

That's it; you now have a completely COPYable *Silicon Dreams* disk. As I said above, after fixing the checksums and the strange sector sequencing, there were no other forms of COPY protection on the disk. However, if you wish to restore a saved game, you must refer to the instruction manual to find a particular word which will allow you to continue play. Needless to say,

error. Note that if you are only ignoring markers, it will work.

I had made complete ProDOS charts previously and used them to write ProDOS IOB 5.25. My thinking was that you could let PLOB do the dirty work, so I didn't bother sending in the charts. However, since Mr. Brett asked for more information, I figured I'd send them anyway. Besides that, my charts proved quite useful to other readers, in particular, I'm talking about COPY+ (Issue #67).

As Mr. Brett noticed, there are blanks for writing address markers, checksums and epilogues and for reading the second data epilogue. And as the RDEXed mentioned, address markers are missing because ProDOS can't format a disk. For whatever reason, ProDOS doesn't read the second data epilogue. This brings up an interesting situation. If the second data epilogue is anything other than a \$AA, ProDOS won't know the difference, but DOS 3.3 copy programs such as COPY+ will fail.

Mr. Brett also lists an error flag in his charts. As it turns out, ProDOS has three separate error flags, one for each of the following:

- [1] address marks, checksum, and epilogues
- [2] data markers
- [3] data checksum and epilogues

As a result of the third flag, only one poke is required for ignoring the data checksum and epilogues, whereas it takes two pokes for the address checksum and epilogues.

There is also a track checker in ProDOS, but I haven't seen any ProDOS disks with this type of protection. Apple modified the device driver between versions 1.0.1 and 1.0.2, and as a result, the track checker was moved two bytes back. I don't know of any other 1.0.x versions, but if you have one, be careful when using this patch.

To make any of the pokes, you must first reference the soft switch at SC089 (49289) twice. That write-enables the language card. Then make the changes, and reference the soft switch at SC08A (49290) to write protect the language card.

Reading protected ProDOS v1.1 - v1.7 disks

Description	Hex	Decimal
Address Marker #1:	SD3AD	54189
Check:	EF	239
Ignore:	00	0
Address Marker #2:	SD3B7	54199
Check:	F2	242
Ignore:	00	0
Address Marker #3:	SD3C2	54210
Check:	E7	231
Ignore:	00	0
Address Checksum & Epilogues:	SD3EA	54244
Check:	D0 15	208 21
Ignore:	18 60	24 96
Track Checker:	SD0A3	53411
Check:	1A	26
Ignore:	31	49
Data Marker #1:	SD43E	54334
Check:	F4	244
Ignore:	00	0
Data Marker #2:	SD448	54344
Check:	F2	242
Ignore:	00	0
Data Marker #3:	SD452	54354
Check:	E8	232
Ignore:	00	0
Data Checksum & Epilogues:	SD4CC	54476
Check:	38	56
Ignore:	18	24

Reading from protected ProDOS v1.0 disks

Description	Hex	Decimal
Address Marker #1:	\$FBAD	64429
Check:	EF	239
Ignore:	00	0
Address Marker #2:	\$FBB7	64439
Check:	F2	242
Ignore:	00	0
Address Marker #3:	\$FBC2	64450
Check:	E7	231
Ignore:	00	0
Address Checksum & Epilogues:	\$FBE4	64484
Check:	D0 15	208 21
Ignore:	18 60	24 96
Track Checker:	\$F8A5	63653
1.0.1 Check:	1A	26
Ignore:	31	49
Data Marker #1:	\$FC3E	64574
Check:	F4	244
Ignore:	00	0
Data Marker #2:	\$FC48	64584
Check:	F2	242
Ignore:	00	0
Data Marker #3:	\$FC52	64594
Check:	E8	232
Ignore:	00	0
Data Checksum & Epilogues:	\$FCCC	64716
Check:	38	56
Ignore:	18	24
Track Checker:	\$F8A3	63651
1.0.2 Check:	1A	26
Ignore:	31	49

Robin Locksley MD

Playing Tip for...

Bard's Tale II

Electronic Arts

To Mr. Tirad: (COMPUTIST #64) about. An equipped Trolling regenerates a characters hit points. A Ring of Power, when equipped, will inflict around 150 points of damage but only to an enemy Magic User. The Soul Mace and the Hawkblade, when equipped, will increase the amount of damage that a Paladin or a Warrior can inflict during battle. The Conjurstaff lowers a Conjurers armor class by 2 points (when equipped of course).

Advanced Playing Technique for...

Bard's Tale II

Electronic Arts

Also to create a Holy Handgrenade that will last forever try the following. Use Copy II Plus 8.3 (copy 3.5) and go to the sector editor. Scan for a characters name. All character information is contained in eight lines of a sector. The characters name is line one. Line six contains the equipment. Each piece of equipment takes up two hex code slots. The Holy Handgrenade code is 01 6D. The first of the two hex codes tells if the item is equipped. 01 means the item is equipped (an 00 means not equipped) 6D is the items call letters in this case a Holy Handgrenade. Line seven contains 16 hex codes but only the last eight are important here. Each code indicates an items amount of usage. Eight items eight codes. If your Holy Handgrenade is a characters third item then change its original code of 01 to FF. Now you have a Holy Handgrenade that lasts almost forever. This works for other items as well. You can also change experience points (line 2) and gold (line 3) while your here. Remember to write all changed hex codes back to the disk. Good luck.

There are two rooms on the second level of the white castle in Bard's Tale 1 that I can't get into. Am I not doing something or am I missing some piece of equipment?

In Computist #62 there is a softkey to get around Bard's Tale 3 code wheel but there is not enough info given. Could we get just a few more hex values before and after the code to be changed?

Is there any way to get into BASIC from Defender of the Crown or do I have to continue to boot my system disk first before I begin play?

And please tell me what APT stands for.

APT stands for Advanced Playing Technique. In other words, a cheat. Usually a cheat that requires you to change the program and/or edit the disk..... RDEXed

Correction: In Computist #60 pg 29. When playing Defender of the Crown if you want to enhance your sword fighting skills don't use the code 03/BC0B that is given. The correct listing should be 03/BC1B.

Brian Hatch WI

Softkey for...

Sensible Speller for ProDOS

Sensible Software, Inc

I recently tried to deprotect Sensible Speller. Naturally I tried to find a softkey in a back issue. I loaded up AppleWorks, placed my Computist Super Index in drive two, loaded a data base file and did a quick records search with openapple R and found seven different references for Sensible Speller. (Anyone who does not own a Computist Super Index by, David R. Hopkins, should certainly get one NOW! See the back of your latest Computist.) Issue no. 16 had a wonderfully documented procedure by Peter Rongays. Although it in itself did not work, I assumed the version I had was only a little different.

Mr. Rongays had you search for the bytes AE 75 A5 BD 89 C0 and later for A9 00 8D 00 A3 8D 01 A3. I found that Mr. Rongays' version was slightly different from my own. The protection, however, was essentially the same. Due to some memory moving, two addresses in the protection were different. The resulting changes are as follows:

Search for:	Change to:
AE 75 A8 BD 89 C0	60 75 A8 BD 89 C0
A9 00 8D 00 A6	60 00 8D 00 A6

The first edit defeats the nibble count. I found it on track 3, sector 5. The second defeats the RAM checksum. I found it on track 3, sector 4. For a detailed account of what changes were made, read issue #16, pages 12-14.

I've been doing serious thinking about the options for the future of COMPUTIST. I feel that a readers wish list is a great idea. That could really get people to encourage new subscribers. I like the tabloid format. Having more information per issue outweighs the lack of a cover. Having a 3-ring binder "book" at the end of the year is also a good idea. However, the most immediate solution is for everyone to recruit new subscribers. Ask COMPUTIST for a flyer and post it somewhere!

I own MultiScribe version 3.00C, copy-

right 1986, 1987 by StyleWare, Inc. Whenever booting it for the first time, it will ask if I have a mouse attached, (I have a //C). After flipping the disk it accesses the disk and hangs. This happened even before I removed the protection. On older //c's and clones, it works fine. Also, if I say I have a mouse attached when I really have a joystick attached and then switch, it works fine. Could someone please help me out!!!

Bill Todd FL

Softkey for...

Kid Niki & Other DataEast Games

DataEast

Requirements:

Blank disk for each game
COPYA (or another copy program that ignores errors)
sector editor

Recently, I purchased Kid Niki and then a friend of mine gave me Ikari Warriors and Commando. I wanted to make backup copies of all the games, but I only had a softkey for Ikari Warriors from COMPUTIST #52, page 13. I sent away for the back issues to copy the other games, but by the time the back issues got here, I had already deprotected the games myself.

On DataEast games, the protection on the disk is held by, or related to the byte "D0" in some way. Search for a sector with a lot of "D0"s on it. If you replace the "D0" with "EA" and the byte after it with "EA" you can deprotect the software. Usually the bytes that you need to change are on track 00. So, now here is how to deprotect two DataEast games by using this method. (Note: When you use this method, don't change all the "D0" bytes to "EA", usually changing 3-5 bytes does it. If the disk will not boot correctly after you tried to deprotect it, try changing some different "D0..." bytes. I had to try several combinations before I found the combination of "D0..." bytes that would deprotect the game.

Kid Niki

1. Copy Kid Niki with any copy program that ignores errors.
2. Run your sector editor and change these bytes

Trk	Sct	Byte(s)	From	To
\$00	\$0E	\$3C-3D ?		\$EA EA
		\$4A-4B ?		\$EA EA
		\$53-54 ?		\$EA EA
		\$5C-5D ?		\$EA EA
		\$7A-7B ?		\$EA EA

That's all folks.

Commando

1. Copy Commando with a copy program that ignores errors.
2. Run your sector editor and change these byte

Trk	Sct	Byte(s)	From	To
\$00	\$06	\$4B-4C	\$D0 F4	\$EA EA
		\$54-55	\$D0 36	\$EA EA
		\$5D-5E	\$D0 2D	\$EA EA
		\$70-71	\$D0 F4	\$EA EA
		\$7B-7C	\$D0 0F	\$EA EA

That's all folks.

Advanced Playing Technique for...

Commando

DataEast

This A.P.T. makes your man on Commando get unlimited lives and grenades. There is something about the bytes "E9 01" on DataEast games that holds a number, like the number of men, grenades, bullets, etc. I found this out from the Ikari Warriors A.P.T. that Willem Moolenaar wrote along with his softkey for Ikari Warriors in COMPUTIST #52. You must find the same number of "E9 01" bytes as limited things on the game and all the bytes must be on the same track. For example, on Commando you have a limited supply of grenades and lives, (that means that there are two things in limited supply on the game) so you need to find two "E9 01" bytes on the same track and sector. Then change the bytes to "EA EA", like this

Trk	Sct	Byte(s)	From	To
\$1C	\$09	\$52-53	\$E9 01	\$EA EA
		\$78-79	\$E9 01	\$EA EA

That's all folks.

Softkey for...

Police Quest

Sierra

Requirements:

4 Blank Disks
a copy program that ignores errors
a sector editor

In order to 'crack' my copy of Police Quest, I did the same thing that I did for Kid Niki and Commando, I looked in my back issues of COMPUTIST to find a softkey for some other Sierra games and then adapted their method to my game. After looking through many softkeys for Sierra brand games, I came across Captain Dan's softkey for King's Quest III in COMPUTIST #61, Pg. 36 and I found some similar-

ties. I searched for the bytes "20 00 FF", like Captain Dan did in the softkey for King's Quest III. I had almost searched the disk when I found them. On track 0D, sector 04, I saw the bytes "20 00 FF". I immediately changed the bytes to "EA EA EA" like the softkey for King's Quest III said to. When I was done, it booted and worked perfectly. So here's the sector edit you make

Trk	Sct	Byte(s)	From	To
\$0D	\$04	\$A5-A7	20 00 FF	EA EA EA

That's all folks! Now you have a deprotected copy of Police Quest.

On all the games that I have deprotected here, I did exactly the same thing, I looked in back issues for softkeys to other games that were made by the same company as my game was. I have found that most software companies use the same, or similar protection on all their games. Even if you are like me and know absolutely nothing about copy protection, you can deprotect many games this way. So far, I've had very good luck using this method. Any time that you can not copy a game, try searching for the same bytes that deprotected another game by the same company. In many cases it will work.

Softkey for...

L.A. Crackdown

Epyx

Requirements:

A copy program that ignores errors
Sector editor (I used Copy II+ v.8.3)

I've been trying to 'crack' L.A. Crackdown for months, with no success. I had tried every softkey I could find which seemed like it may 'crack' L.A. Crackdown. Finally, in a last resort I saw the Dive Bomber softkey by Jim Hart in issue 66. I read track 00, sector 09, like it said in the softkey, and the bytes matched almost exactly. All but 3 or 4 bytes were the same. I immediately changed the bytes to the same thing that the bytes on Dive Bomber were change to, and it worked. I had deprotected L.A. Crackdown with a little help from Jim Hart in COMPUTIST #66.

Here's what you do to deprotect L.A. Crackdown.

1. Copy both sides of L.A. Crackdown with a copy program that ignores errors.
2. Run your sector editor and change these bytes on the copy you just made.

Trk	Sct	Byte(s)	From	To
0	9	10-2B	A6 2B BD 89 C0 BD 8E C0 A9 80 85 FD C6 FD F0 71 20 AF F0 B0 6C A5 F9 C9 00 D0 F1 A0	A9 E7 85 F8 A9 FC 85 F9 85 FF A9 EE 85 FA 85 FD 85 FE A9 F3 85 FB A9 70 85 FC D0 50

Now that you are done with that, you have deprotected L.A. Crackdown. Now before you drop that magazine and start to play your backup copy, I have a few hints and suggestions to make your 'cracking' easier. First of all, a novice can deprotect software thanks to COMPUTIST. I am a super novice. I only know BASIC and a few principles for Machine Language, (like how to count in Hex). I have successfully deprotected Police Quest, L.A. Crackdown, Kid Niki, and a few other games. I used a few easy steps to deprotect these games.

1. Find the manufacturer of the software you want to deprotect.
2. Find as many COMPUTIST softkeys for deprotecting software by the same manufacturer that manufactured your software as you can.
3. Try out as many of these softkeys as possible until you find one that works. (Note: a lot of times a softkey for one program is almost, but not quite the same as another softkey for a program by the same manufacturer. Use common sense to find the right softkey, that's all I do.)
4. Send the new softkey into COMPUTIST so other people can use the softkey.

I hope you can benefit from this information.

James J. Harvey MI

Softkey for...

Addison-Wesley Mathematics Skills Software Series

Addison-Wesley Publishing Company

Requirements:

DOS 3.3 System Master
Copy II+ or other with a Sector Editor

A short time ago, a friend of mine who is a school teacher (I am a retired teacher), brought some math programs to me and ask if I could deprotect them. He said that he was really concerned for the safety of the programs since the computer accesses the disk at various times during the program's operation, and his students have, at times, been rather careless with the disks.

The Addison-Wesley Mathematics Skills Software Series levels 3, 4, 5, 6, 7, and 8 deals with building math skills for early and later Elementary grades.

The Level 7 and 8 programs have two disks, the main one of which is double sided. The Level 7 program has also included a calculator which may be called up onto the screen at any time. A teacher menu and student records are included in all of the programs, and teacher and student records may be printed.

All efforts to copy these programs using EDD4, Locksmith, and Copy II+ failed. However, here is a solution to the problem of making backup copies.

1. Write protect the original program disk.
2. Boot DOS 3.3 System Master.
POKE 47426,24 ignore checksum and epilg errors
RUN COPYA.
3. Copy the original program onto a blank disk.
4. Boot Copy II+ or your favorite sector editor. Scan for BD 8C C0 10 FB C9 E7 D0. I found it on track \$00, sector \$05, starting at byte \$73, and at byte \$7C.
5. Sector edit the following:

Trk	Sci	Byte(s)	From	To
\$00	\$05	\$79	\$E7	\$AA
		\$82	\$E7	\$AD
		\$AC	\$38	\$18

That's it. You now have a deprotected copy. If your original program disk is two-sided, you may straight copy the back side since the back side is not protected. This should work for other programs in the Addison-Wesley Mathematics Series.

Softkey for... **Rainbow Painter Stickers**

Springboard Software, Inc

Clay Harrell (Vol. 63 P. 17-18) spent a considerable amount of time and effort in the development of his softkey for Rainbow Painter and Stickers. His article was of great interest to me since I had just had a visit from another school teacher friend of mine who wanted a deprotected copy of both of these programs.

I checked both of my friends programs for the codes that were in Mr. Harrell's softkey. In Stickers, I found this code only once on track \$08, sector \$01, starting at byte \$92. In Rainbow Painter, I found this sequence only once on track \$09, sector \$07, starting at byte \$79. Making the changes to my copies as he suggested did not work. Obviously the versions of my programs were different than the ones he has.

Here is my approach to the problem which was much easier to solve.

1. Boot a DOS 3.3 System Master disk.
POKE 47426,24 ignore checksum and epilg errors
2. Copy the original program onto a blank disk
3. Use Copy II+ or your favorite sector editor and sector edit the following:

Trk	Sci	Byte(s)	From	To
\$00	\$03	\$42	\$38	\$18

That's it. You now have a deprotected copy of Stickers and Rainbow Painter.

One last parting remark before I close is a plea for some help. Does anyone know where I may obtain the documentation for The CIA? I would be grateful for the information.

Daniel Bashford Australia

Playing Tip for... **Wings Of Fury** *Broderbund*

When you are on Aircraft Carrier press esc and type ASK, you will receive unlimited aeroplanes. Also on the IIGs try playing in Fast mode, its great fun!, except when you land you have to change it back to normal again!

Playing Tip for... **Leisure Suit Larry** *Sierra OnLine*

Hold down ctrl X while booting to bypass the questions, it is possibly option X on the IIGs. Also when you are at the telephone, dial Sierra for 5 bonus points!

Playing Tip for... **Defender Of The Crown (IIGS)** *Mindscape*

Hold down the option key while booting, this does something apparently, I don't know what though.

Playing Tip for... **Defender** ?

When your Score is 20000 an Easter egg appears. Get it and you will get 20000 points.

Playing Tip for... **Bards Tale III** *Electronic Arts*

To get your experience and gold up high, you must first have a spare slot in your party. Go to the

old man and, before going in, cast a demon, or any illusion, etc. Then go in and he will award you about 2500000 bonus from memory, and some gold I think. Then exit from there and drop the demon from the party, then cast another one and enter again, he will award you again. You can do this infinite times but it gets too easy and VERY boring after doing it 150 times!

Playing Tip for... **Print Shop Companion** *Broderbund*

Try typing "Steven" or "Stevie", the screen will do a mirror flip, also on the back side of the disk type it, there is a game called DRIVER that will hopefully run.

Jack Moravetz OH

Softkey for... **Math Blaster Plus Homeworker Math And Me**

Davidson & Associates, Inc.

Math Blaster Plus is a good teaching tool for basic math facts. It helps students master addition, subtraction, multiplication, division, fractions, decimals, and percents. There are over 750 math facts presented in various levels of difficulty and there are review lessons. A student can have a certificate printed for obtaining a perfect score on the exercises.

Homeworker has an Appleworks type look about it. It offers the teacher or student a Textwriter, Outliner, Flashcard, Calendar, and Gradekeeper.

Math and Me is an early education math program for shapes, patterns, and addition.

All three of these fine programs had similar copy protection. The changes I made are to the ProDOS file. Maybe someone else can find a better way. It doesn't matter which version of ProDOS you use, although it's a good idea to use a later version on Homeworker. It came with the infamous ProDOS 1.1.1 on it.

1. Copy the disk with any disk copier. If you get a read error on track \$22 with Homeworker, use Locksmith 6.0 or COPYA with the DOS error checking disabled. It's possible to copy the files from either disk, but the volume name is critical. Math Blaster Plus boot side is named /MATH.0 and side two is named /MATH.1. Homeworker is named /HOMEWORKER.
2. Using a sector editor scan for the bytes A5 42 C9. I found them on track \$04, sector \$07, byte \$D0.

Here is what you should find:

A5 42 C9 04 B0 10 A5 46
A6 47 8E 56 D3 F0 09 CA
D0 04 C9 18 90 02 38 60
18 60 00 00 00 00 00 00
00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00

3. Change the A5 42 C9 to 4C EA D6 and add these bytes after the 18 60 bytes at byte SEA: A5 47 C9 01 D0 E6 A5 46 C9 10 D0 E0 4C E6 D6

Here is how it should look now:

4C EA D6 04 B0 10 A5 46
A6 47 8E 56 D3 F0 09 CA
D0 04 C9 18 90 02 38 60
18 60 A5 47 C9 01 D0 E6
A5 46 C9 10 D0 E0 4C E6
D6 00 00 00 00 00 00 00

4. Write the sector back to the disk.

Remember to make these changes if you update to a later version of ProDOS because the changes are to ProDOS. This may softkey other Davidson ProDOS releases as well. It will not work with Read N Roll.

Softkey for... **Grammar Examiner** *DesignWare/Britannica Software*

Thanks to Clay Harrell's fine article on deprotecting DesignWare programs in COMPUTIST #18 I was able to softkey Grammar Examiner. Grammar Examiner was written in MicroMotion FORTH-79 so don't bother trying to find any files on the disk. Clay's article gives a good explanation of what the protection is and how to bypass it. In case you missed that one, here is enough to softkey the disk.

1. Copy the disk with any disk copier.
2. Using a sector editor, make the following changes:

Trk	Sci	Byte(s)	From	To
\$00	\$05	\$39	00 BB	93 B7

3. Write the sector back to the disk.

Softkey for... **Magic Slate II** *Sunburst*

Magic Slate II is a word processor that supposedly has virtual memory capability. It has an 80 column version and a 40 column version as

before. I've heard that it even comes in a 20 column version for young children although I haven't seen it.

The protection on Magic Slate II was a nibble count contained in the system startup file.

1. Copy the disk using any disk copier.
2. Using a sector editor, scan for BD 8C C0. They may appear in several places. The bytes containing the nibble count routine were found on track \$04 sector \$0F.
80 column version at byte \$38
40 column version at byte \$25
3. 80 column version: Change the bytes beginning with the BD 8C C0 at byte \$38 and ending with byte \$4F to EA (24 bytes total).
40 column version: Change the bytes beginning with the BD 8C C0 at byte \$25 and ending with byte \$3C to EA (24 bytes total)
4. Write the sector back to the disk.

Softkey for... **Voyage of the Mimi**

Holt, Rhinehart, and Winston/Bankstreet College Project

The Voyage of the Mimi is an interesting software and video cassette series for teaching map and navigation skills. The series contains several modules. The two I have are Maps and Navigation and Introduction to Computing.

The copy protection consisted of having one track that a standard copier would not read and a nibble count. It is fairly easy to make a bit copy of this program, but it is just as easy to softkey it.

The protection was the same routine for each disk in the series. It was located in different places on two of the disks.

Here is the protection routine that I found:

BD 8C C0 LDA \$C08C,X
10 FB BPL \$0E59
C9 FF CMP #\$FF
D0 F7 BNE \$0E59
BD 8C C0 LDA \$C08C,X
10 FB BPL \$0E62
C9 FF CMP #\$FF
D0 EE BNE \$0E59
BD 8C C0 LDA \$C08C,X
10 FB BPL \$0E6B
C9 FF CMP #\$FF
F0 F7 BEQ \$0E6B
C9 95 CMP #\$95
D0 E1 BNE \$0E59
A0 C0 LDY #\$C0
BD 8C C0 LDA \$C08C,X
10 FB BPL \$0E7A
88 DEY
D0 F8 BNE \$0E7A
BD 8C C0 LDA \$C08C,X
10 FB BPL \$0E82
C9 CA CMP #\$CA
F0 03 BEQ \$0E8E
4C 00 C6 JMP \$C600
BD 88 C0 LDA \$C088,X
60 RTS

When the protection routine was encountered on the copy, the disk would sit and spin in the drive or reboot. The best way to eliminate the protection was to not let the program get there. I placed a RTS at the first byte of the routine and the copy booted and ran fine.

1. Using a disk copier that will ignore errors, copy all the disks. I used Locksmith 6.0.
2. Using a sector editor and starting at track \$01, scan for the bytes BD 8C C0.

Where the protection routine is found:

Maps and Navigation — Disk 1, side A: Pirate's Gold, Lost at Sea. Disk 1, side B: Hurricane. Introduction to Computing — Disk 2, side A: Turtle Steps. Disk 2, side B: Turtle Steps.
Track \$01, sector \$0C, byte \$59

Maps and Navigation — Disk 2: Rescue Mission.
Track \$01, sector \$0C, byte \$C9

Introduction to Computing — Disk 1: Whale Search.
Track \$01, sector \$0A, byte \$59

3. Change the BD 8C C0 in these sectors to 60 8C C0 and write them back to the disk.

To Kay Jun's request about Microcomputer Workshop Courseware. I've been able to deprotect many of their programs by using Super IOB and the Swap Controller. Sometimes just disabling headers and trailers won't give you as reliable a copy as Super IOB. Several of the programs that were listed can be softkeyed using the Swap Controller.

A BUG in Teacher's Tool Kit Series

Also in regard to The Teacher's Tool Kit Series by Hi Tech that was in COMPUTIST #65, I made a serious omission in my softkey. They seem to have some type of device driver in their ProDOS that won't allow you to prefix any slots with 5 1/4" drives. What I did was prefix slot 7 which had my 3.5" drive. Their ProDOS doesn't prohibit prefixing the 3.5" drive. From there I formatted my Ramworks III with RAM.DRV from Glen Bredon's ProSEL and used Cat Doctor to copy the files from the protected disk to the /RAM. I then rebooted from slot 7 and ran RAM.DRV (Otherwise you cannot format a

normal disk). I used Cat Doctor to format a blank with the same volume name as the original protected disk and copied the files from the /RAM to the 5 1/4 disk. This is the same method I used for copying Sunburst ProDOS releases. The only problem is finding a disk that has BASIC.SYSTEM on it. You can tell by the BASIC prompt that appears while it's booting. Once you interrupt the boot at the BASIC prompt by hitting control/reset, you can CATALOG the protected disk and use that BASIC.SYSTEM to prefix a 3.5" drive and copy all of Sunburst's ProDOS disks. This method is for those Sunburst releases that won't copy with a normal disk copier.

Bob Dylan OR

Softkey for... **Where in the World is Carmen Sandiego (GS)**

Broderbund

The disk copies without any block errors, so I scanned for a nibble count, and this is what I found on block \$9:

1151:6B	RTL		
1152:08	PHP		this section begins at 01/182D in monitor
1153:E2 30	SEP #30		set 8-bit accumulator
1155:A2 20	LDX #20		nibble count begins: track 20
1157:A0 01	LDY #01		side 1
1159:8B	PHB		
115A:4B	PHK		
115B:AB	PLB		
115C:20 5C 02	JSR 025C		jump to 00/119C below for "count-m-up" branch if bad
115F:D0 23	BNE 1184		
1161:8A	TXA		
1162:8D 95 02	STA 0295		store flag to insure this code has been run
1165:98	TYA		
1166:8D 96 02	STA 0296		store another flag
1169:0D 95 02	ORA 0295		
116C:F0 16	BEQ 1184		branch if bad
116E:A2 21	LDX #21		nibble count on trk 21
1170:A0 01	LDY #01		side 1
1172:20 5C 02	JSR 025C		jump to 00/119C below to count-em-up
1175:D0 0D	BNE 1184		branch if bad
1177:8A	TXA		
1178:8D 98 02	STA 0298		store a flag to insure this code has been run
117B:98	TYA		
117C:8D 98 02	STA 0299		store another flag
117F:0D 98 02	ORA 0298		
1182:D0 13	BNE 1197		branch if good
1184:48	PHA		
1185:A9 33	LDA #33		
1187:8D 95 02	STA 0295		
118A:8D 96 02	STA 0296		
118D:8D 98 02	STA 0298		
1190:8D 99 02	STA 0299		
1193:68	PLA		
1194:82 02 00	BRL 1199		
1197:A9 00	LDA #00		everything passed
1199:AB	PLB		
119A:28	PLP		
119B:6B	RTL		

Immediately below there follows:

119C:5A	PHY		
119D:DA	PHX		
119E:F4 85 02	PEA 0285		
11A1:F4 85 02	PEA 0285		
11A4:22 C1 00 04 JSL	0400C1		jump to Block 4E, 00/1172 = actual count routine
11A8:8D 93 02	STA 0293		store returned value
11AB:68	PLA		pull all extra values
11AC:68	PLA		
11AD:68	PLA		
11AE:68	PLA		
11AF:68	PLA		
11B0:68	PLA		
11B1:AD 93 02	LDA 0293		get returned value
11B4	RTS		

Then follows a small section of code where the stored flags are pulled:

11B5:8B	PHB		this section begins at 01/1890 in monitor
11B6:4B	PHK		
11B7:AB	PLB		
11B8:AD 95 02	LDA 0295		
11BB:AB	PLB		
11BC:6B	RTL		
11BD:8B	PHB		this section begins at 01/1898 in monitor
11BE:4B	PHK		
11BF:AB	PLB		
11C0:AD 98 02	LDA 0298		
11C3:AB	PLB		
11C4:6B	RTL		

A search through the monitor finds the original routine that jumps to the nibble count and flag checks, located at 03/4D22 in the monitor:

4D22:22 2D 18 01 JSL	01182D		Jump to nibble count
4D26:22 90 18 01 JSL	011890		Jump to code that pulls flags
4D2A:85 EA	STA EA		Temporarily store the flag at EA
4D2C:22 98 18 01 JSL	011898		Jump to more code that pulls flags
4D30:85 E8	STA E8		
4D32:A5 EA	LDA EA		

This routine continues quite a ways, ending in an RTL. By moving the RTL up to the front of

this routine at 03/4D22, the protection is completely bypassed. To make the change on disk:

Blk	Byte(s)	From	To
\$67	\$22	0B	6B

Everett B. Young CA

Bitkey for...

Math Blaster Mystery

Davidson & Associates

The Algeblaster and Homeworker bit copiers offered by Gerald E. Myers in Computist #64 also work for Davidson & Associates' new program, Math Blaster Mystery. Specifically,

1. Format a disk for ProDOS. Give it a temporary volume name, /M.
2. Copy a current version of ProDOS, such as the one that comes with Copy II Plus, to the new disk.
3. Copy the files from Math Blaster Mystery, except for ProDOS, to the new disk.
4. Using Copy II Plus, Bit Copier, Manual Bit Copy: Start Track \$22, End Track \$22, No Synch (default), No keep track length (default); hit / for "To Modify" menu. Change parameter 65 to 00; press ESC; press RETURN to copy Track \$22.
5. Rename Volume /M to /MBM.
6. Use Copy Disk to copy side 2 of Math Blaster Mystery.

Softkey for...

Math Blaster

Davidson & Associates

Requirements:

Controller for Spell It! from Computist #58
Super IOB 1.5
Blank disk
Diversi-DOS
DOS 3.3
Copy II Plus

Edward Teach's softkey for Math Blaster, also from Davidson, in Computist #61, included instructions beyond my grasp. And my copy of the program varied slightly from his. So I experimented, using Mr. Teach's ideas plus the Spell It! controller and tips provided by Dick Lavallee in Computist #58. The following alternate method deprotects my Math Blaster program disk:

1. Type in the controller for Spell It!. Check it carefully, then SAVE CON.SPELL.IT.58 on a Super IOB disk.
2. Boot Super IOB, and choose CON.SPELL.IT.58. Put a blank disk in Drive 2, then follow the prompts, answering Y to Format Back Up First, until the COPY DONE message appears.
3. Boot Copy II Plus (I don't have the mythological COPYA.) and choose Copy -DOS to copy Diversi-DOS to the new disk. Normal DOS 3.3 doesn't seem to work.
4. Choose Rename File on Drive 2 to rename H(ctrl Z)ELLO to HELLO. The ctrl Z in the file name only appears with the Catalog w/Hidden Characters option.
5. Use Copy Disk to copy side 2 of Math Blaster to the back of the new disk (after notching it).
6. Boot normal DOS 3.3 in Drive 1. Flip the new disk in Drive 2 back to side 1.

LOAD HELLO,D2

POKE104,32

POKE33,33

LIST400

Change the = to <> and the <> to =. Line 400 now reads 400 IF PEEK(40324) <> 173 OR PEEK (47094) = 0 THEN 1000.

POKE104,8

SAVE HELLO

LOAD M(ctrl Z)ATH BLASTER

LIST359

Change the = to <>. Line 359 now reads 359 IF PEEK (40324) <> 173 THEN 403.

LIST399

Change the = to <> and the <> to =. Line 399 now reads 399 IF PEEK (47094) = 0 OR PEEK (40324) <> 173 THEN 403.

SAVE M(ctrl Z)ATH BLASTER

Now comes the part that may put off some beginners. Use Copy II Plus Sector Editor to search for AD E2 28 34 30 33. In the first location, Trk \$08 Sct \$03 Byte \$6A, change AD to B2. Mr. Lavallee explains that this changes an Applesoft IF to REM. The same sequence may be found later, on track \$13, but do not change it. Apparently, line 400 of HELLO must remain an IF. Follow these step-by-step instructions (written for Copy II Plus.8.x):

Boot Copy II Plus again and choose Copy - Bit Copy 5.25 (press <return> to continue).

Press 6 for Slot Number

Choose Sector Editor

Press 2 to choose Drive 2, which contains side 1 of the new disk

Press S for Scan

Press H to Scan for Hex

Type AD E2 28 34 30 33 and press return

Drive 2 whirls, looking for the code. Eventually it stops with the cursor on 33.

Press J five times to back up to AD

Press H to Enter Hex Values

Type B2 return

Press esc

Press W return return to write the new value to disk

Press esc to get out of Sector Editor

Put write-protect tabs on both notches before using the new disk.

Make any additional backups of Math Blaster from the newly deprotected copy, using Copy II Plus, Copy Disk. Be sure to put write-protect tabs on the notches before using the copies.

Gary Chan Keng Hwee Singapore

Singapore Microcomputer Society Apple User Group

Softkey For...

Arkanoid II: The Revenge of DOH (GS)

Taito

Requirements:

512K IIGs
3.5" disk copier
3.5" block editor

Arkanoid II for the GS is one of the best arcade games that I have come across. I have been playing it almost every night.

The program is heavily protected but I managed to deprotect it. It can be copied easily now and, also, can be transferred to a harddrive for faster disk access. The protection is a 'check track length' technique, simple but effective. After a little hacking, I found that the program calls the check routine 5 times throughout the different stages of the game. I modified the program to ignore the check routine.

Blk	Byte	From	To
32A	0A2	22	AF
	0A6	90 09	80 0F
	182	22	AF
	18A	D0	80
33A	142	90	80
353	15D	22	AF
	161	90	80
391	14F	22	AF
	153	90	80

Softkey for...

Bubble Ghost (GS)

Taito

Requirements:

512K IIGs
3.5" disk copier
3.5" block editor

To play Bubble Ghost, you are required to refer to the manual and enter the level number of the screen shown. But the manual wasn't very clear and kinda difficult to read so I came up with this patch to bypass the screen check.

Blk	Byte	From	To
1A7	49	AF	22
	55	03	00
	57	FF 01	01 02

Softkey for...

Xenocide (GS)

Micro Revelations

Requirements:

512K IIGs
3.5" disk copier
3.5" block editor

Xenocide is another great game for the IIGs. It's fast and it features good graphics and stereo sound. The manual states that you can transfer the game to a harddrive but you will still need your 3.5" drive for verification. I copied the game with Copy II Plus and it worked! I was surprised that they didn't protect it. I was wrong, the program uses custom skewing on the disk. Some of you may know that our 3.5" disks are skewed with a factor of 2:1 or 4:1. A custom skewing is a mild form of protection but they went further. I patched the program to skip the check and it quit working, the program won't allow you to proceed to cave II. It took a lot more hacking before I overcame this protection.

Blk	Byte	From	To
042	19E	8B 4B AB	82 C9 00
043	06D	06 00	46 1E
	073	E2	C2
	076	60	6B
634	07D	8B 4B AB	82 C9 00
	14C	06 00	46 1E
	152	E2	C2
	155	60	6B

Doyle H. Brown PA

IIGs Softkey for...

Duel

Accolade

Duel is an auto race game for the IIGS with excellent graphics. It uses track 20/21 nibble count protection. Searching for A2 20 A0 01

finds the subroutine. An RTL at the beginning bypasses it.

Blk	Byte(s)	From	To
\$16E	\$123	08	6B

Dick Lavallee MA

Softkey for...

Read 'N Roll 1.1

Davidson

Requirements:

One blank 3.5" disk
ProDOS block editor for 3.5" disk
3.5" Disk copier that ignores errors (Copy II+) or
Two blank 5.25" disks
ProDOS block editor for 5.25" disk
5.25" Disk copier that ignores errors [Copy II+]

Read 'N Roll comes on one 3.5" disk or two 5.25" disks (4 sides). The 5.25" program disk is has an error on block \$110 and the 3.5" disk has an error on block \$308.

The protection code is in the text file named PR0. I reversed the code 08 03 [\$0308] to 03 08 [\$0803] so that there would always be a block error read when the program tries reading block \$803 which is out of range on a 3.5" disk. And, similarly, reversed the code 10 01 [\$0110] to 01 10 [\$1001] which is out of range on a 5.25" disk.

1. Copy disk or disks.
2. Edit the Program Disk only [5.25"]

5.25" disk

Blk	Byte(s)	From	To
\$F5	\$12E-12F	10 01	01 10

If the From bytes at block \$12E are not found at this location, search the disk for 10 01 6B 51 71 52 and reverse the first two bytes from 10 01 to 01 10.

3.5" disk

Search the disk for 08 03 6B 51 71 52 and reverse the first two bytes from 08 03 to 03 08.

Both protection codes are in the same text file PR0. Your done!

Softkey for...

Solving Quadratic Equations

Micro Computer Workshops Courseware

Requirements:

COPYA or similar copier.
Blank disk.
ProntoDOS or similar fast DOS or Master Create on the DOS 3.3 System Master Disk.

Ignore the disk drive chatter (Kay Jun, Computist #66/12) because of some unreadable sectors.

1. Run COPYA and press reset at the menu.

70

Call-151

B954:29 00

B942:18

B990:29 00

B99A:29 00

BE48:18

ctrl C

RUN

2. Place Pronto DOS or other DOS with boot-up file HELLO.

Done!

Softkey for...

Robomath

Mindplay

Requirements:

Copier that ignores errors
Blank disk
Sector editor

Robomath contains an unreadable track (\$1D) and the deprotection permits its use on the IIGs without the "damaged of pirate copy" display.

1. Copy disk.
2. Sector edit:

Trk	Sct	Byte(s)	From	To
\$01	\$01	\$07-08	A9 08	18 60

If the From bytes are not found at this location, search the disk for A9 1D 8D 2C B7 and change A9 1D to 18 60.

The Nipper Canada

Softkey for...

Writing & Publishing Center (IIG IIGs)

Learning Co.

Requirements:

Blank 3 1/2" disk
Copy program that ignores errors
Sector editor

This is a program that no primary school should be without and the 3 1/2" version requires no disk swapping making it even easier to use. The protection, which checks for a bad block

(#7), is called from the file named X. The initial code is as follows:

20 37 70	JSR 7073	execute the check
B0 1F	BCS b+F)	check error
A0 02	LDY #02	set counter
B9 67 75	LDA 7567,Y	get byte
C9 01	CMP #01	compare
D0 0C	BNE }+0C}	branch if error
C8	INY	increment counter
B9 67 75	LDA 7567,Y	get byte
C9 00	CMP #00	compare
D0 04	BNE }+04}	branch if error
18	CLC	clear carry
A9 00	LDA #00	0 in accumulator.
60	RTS	return OK

The protection can be nullified by simply clearing the carry, branching over the code and returning to the program with a zero in the accumulator. This is done by replacing the JSR 7037 (20 37 70) with a CLC and a BCC 14 (90 14). In cookbook fashion:

1. Make a copy of your 3 1/2" Writing & Publishing Center disk.
2. Edit the copy as follows.

Blk	Byte(s)	From	To
\$16E	\$132	E2 30	80 3C

Softkey for...

Test Drive II (IIGs)

Accolade

Requirements:

Blank 3 1/2" disk
Copy program
Sector editor

Test Drive II has become the newest addictive game around the house. The play is smooth and the graphics are fabulous. If you don't have one yet this is a must get program. The disk is however protected and has a lot of disk access. The protection is almost the same as on other recent Accolade releases and can be located by scanning the disk for the bytes A2 20 A0 01. On my disk they are found on block \$16E. Starting at byte \$134 the code is as follows:

34 E2 30	SEP #30	8 bit accumulator.
A2 20	LDX #20	track 20
A0 01	LDY #01	side 1
20 40 C5	JSR C540	disk check
8E 97 C5	STX C597	store result
8C 98 C5	STY C598	store result
41 C2 30	REP #30	16 bit accumulator.
AD 97 C5	LDA C597	get result
93 6C 20	CMP #206C	check value
90 29	BCC (+29)	error if less
C9 02 21	CMP #2120	check value
B0 24	BCS (+24)	error if more
50 E2 30	SEP #30	8 Bit accumulator.
A2 21	LDX #21	track \$21
A0 01	LDY #01	side 1
20 40 C5	JSR C540	disk check
8E 97 C5	STX C597	store results
8C 98 C5	STY C598	store results
5F C2 30	REP #30	16 bit accumulator.
AD 97 C5	LDA C597	get result
C9 B0 1D	CMP #1DB0	check value
90 0B	BCC (+0B)	error if less
C9 78 1E	CMP #1E78	check value
B0 06	BCS (+6)	error if more
6E C2 30	REP #30	16 bit accumulator.
AB	PLB	
7A	PLY	
28	PLP	
6B	RTL	return OK

As you can see if all goes well you are returned to the program by the RTL. The deprotection was very simple in that all you have to do is jump over the code by replacing E2 30 with the branch always instruction 80 3C. This just jumps past the last error check. In cook book fashion.

1. Make a copy of your Test Drive II.
2. Edit the copy as follows.

Blk	Byte(s)	From	To
\$16E	\$132	E2 30	80 3C

If you want to put it on your hard disk you will have to scan the disk and replace the master disk name (TD2MASTER) with the hard disk path where you put the game. Enjoy!

B. Dudley Brett Canada

Super 6.0 FastcopyA

Scott M. Simon's article on Super COPYA v1.1 (COMPUTIST v.66, p.22) introduced a novel twist to the process of patching RWTS to be able to read protected software. Instead of remembering or looking up all those patches, why not have a program to do the patching for you? Scott's approach stimulated me to research further a sector copy program familiar to COMPUTIST readers, Locksmith 6.0 Fastcopy, in order to accomplish the same for it as had been accomplished for COPYA.

At one time, I had the idea that other copy programs such as LS 6.0 Fastcopy could be used in a similar way to COPYA. ie. Place patches in DOS to turn off error traps when a sector read is made, then run the copy program. However, when I tried this with Locksmith, Disk Muncher or other sector copiers, it did not work. Apparently, these programs use their own RWTS, so

patches to DOS will not accomplish anything. I did nothing further until I saw Scott's article. Then I realized that if the same could be done for Locksmith that I would obtain a great utility. First, it would copy much faster than COPYA. It would also display sectors as read and written, and not "hang" upon reaching an unreadable sector. Further, from comments in COMPUTIST from recent subscribers, COPYA is becoming hard to find, especially for those who purchased an Apple recently. Finally, other interested individuals may apply my approach to other sector copy programs in order to make copy deprotection less tedious.

We have Jerry Lewis(!) to thank for first releasing Locksmith 6.0 Fastcopy to the COMPUTIST world (v.43, pp.12-14). His technique to produce a standalone file (LS 6.0 Fastcopy) inspired me to design a fastloading disk, booting this copied program (5 Second Boot into Locksmith 6.0 Fastcopy - v.56, pp.12-13). Others have suggested modifications in order to make the program work on the IIGs. It is apparent that the copy program is popular to all.

I started my research by looking LS 6.0 Fastcopy and scanned through it looking for nibble reads. Any accessing of \$C08C would show this. Fortunately, I quickly found the routine to read the address field from \$2344 to \$239C and the data field from \$239D to \$240B. It was similar to routines in DOS 3.3, with an interesting difference. Instead of directly comparing nibbles (C9 D5 CMP #D5), Locksmith places the read marker bytes in a lookup table from \$00E8 to \$00F2 and uses compares to the bytes held at an address in memory. Thus, C5 E8 CMP #E8 would ask to compare the nibble read with the byte held at address \$00E8, which just happens to be D5. These marker read bytes, I found originally contained in Locksmith from \$34CC to \$34D6. Immediately after, starting at \$34D8 are the marker write bytes. Apparently, these are all moved to page \$00 when the program is first run. Other than this, I found nothing vastly different from the normal DOS 3.3.

Referring to Bill Jetzer's excellent article showing patches to DOS 3.3 (and DOS 3.2) in order to read protected disks (v.60, pp.10-13), I decided to find similar patches in Locksmith and then write an Applesoft program in the style of Scott Simon, allowing one to fully unlock the power of Locksmith. My first attempts succeeded so well that I decided to enlarge the scope of the program. First, I thought it would be valuable to allow one to use either the Locksmith Fastcopy or the COPYA option. Some people may have only one of these two programs. Secondly, I decided to allow the capability to change the write bytes in RWTS. This would allow one to protect a disk, rather than the usual deprotection process.

The accompanying Applesoft listings are the result. Enter listing 1 and save it as LOADER.S6FCA. Enter listing 2 and save it as SUPER.6.0.FASTCOPYA. On the same disk you should also have the following files: LS 6.0 FASTCOPY, COPY.IOB0 and COPYA.REV. This last file is just good old COPYA with three lines added. See Listing 3. Simply Load COPYA, add the three lines given and save as COPYA.REV.

Listing 1 - LOADER.S6FCA

```

10 REM *****
20 REM * SUPER 6.0 FASTCOPYA *
30 REM * LOADER *
40 REM * BY B.D.BRETT *
50 REM * JUNE 27, 1989 *
60 REM *****
70 TEXT : HOME
80 D$ = CHR$(4)
90 VTAB 1: HTAB 10: PRINT "SUPER
6.0FASTCOPYA"
100 HTAB 10: PRINT "ARRANGEDBY
B.D.BRETT"
110 HTAB 14: PRINT "JUNE27,1989"
120 VTAB 6: HTAB 5: INVERSE :
PRINT "CHANGESFORLSORDOS3.3
RWTS" : NORMAL
130 VTAB 9: HTAB 3: PRINT "Thisis
anutilitytoallowyouto"
140 PRINT "modifythelocksmith
6.0FastcopyRWTS"
150 PRINT "orDOS3.3RWTSinorder
to"
160 PRINT "readfromandwriteto
protected"
170 PRINT "software.UseaNibble
Editortofind"
180 PRINT "anyaddressordata
fieldchangesin"
190 PRINT "adisketteandnote
whicherrortraps"
200 PRINT "needtobeabortedin
DOS.Makethese"
210 PRINT "changes(nextpage)
prior tousing"
220 PRINT "locksmithorCopyA,but
be wisetoyour"
230 PRINT "choices."
240 VTAB 22: PRINT "PRESSANYKEY
TOCONTINUE" : GET Z$: PRINT
250 HTAB 3: HTAB 3: PRINT "Doyou
wishlocksmith(L)or"
270 VTAB 5: HTAB 3: PRINT "Doyou
wishCopyA(C)?" : GET
X$: PRINT

```

```

280 IF X$ < > "C" AND X$ < > "L"
THEN 260
290 FL = 1: IF X$ = "C" THEN FL =
2: POKE 16384,0
300 POKE 16384,FL
310 IF X$ = "C" THEN PRINT D$;
"RUNCOPYA.REV"
320 PRINT D$; "BLOADLS6.0FAST
COPY"
330 PRINT D$; "RUNSUPER6.0FAST
COPYA"

```

Checksums

10-\$BADD	130-\$F9E8	230-\$8BDC7
20-\$9B13	140-\$F9F8	240-\$8995
30-\$4D3B	150-\$2CE9	250-\$F645
40-\$AD92	160-\$0A54	260-\$4757
50-\$C899	170-\$3CC1	270-\$9EA0
60-\$FF65	180-\$E639	280-\$B744
70-\$4263	190-\$6B35	290-\$DCB9
80-\$991D	200-\$B052	300-\$DABF
90-\$5DEE	210-\$DA74	310-\$BD24
100-\$56F5	220-\$12AF	320-\$B305
110-\$3AE0		330-\$9AA3

Listing 2 - SUPER 6.0 FASTCOPYA

```

10 REM *****
20 REM * SUPER 6.0 FASTCOPYA *
30 REM * BY B.D.BRETT *
40 REM * JUNE 27, 1989 *
50 REM *****
60 TEXT : HOME
70 FL = PEEK (16383): IF FL < > 1
AND FL < > 2 THEN PRINT CHR$(
4); "RUNLOADER.S6FCA"
80 REM LS 6.0 (FL=1), COPYA (FL=2)
90 DIM A$(12,2),A(12,5),B$(11),
B(11,3)
100 VTAB 1: HTAB 10: PRINT "SUPER
6.0FASTCOPYA"
110 HTAB 10: PRINT "ARRANGEDBY
B.D.BRETT"
120 REM LS 6.0 (FL=1), COPYA
(FL=2)
130 FOR I = 1 TO 5: READ
A$(I,1),A$(I,2),A(I,1),A(I,2),
A(I,3),A(I,4),A(I,5): NEXT
140 READ A$(6,1),A$(6,2),A(6,3),
A(6,4),A(6,5)
150 FOR I = 7 TO 11: READ
A$(I,1),A$(I,2),A(I,1),A(I,2),
A(I,3),A(I,4),A(I,5): NEXT
160 READ A$(12,1),A$(12,2),A(12,3),
A(12,4),A(12,5)
170 FOR I = 1 TO 11: READ B$(I),
B(I,1),B(I,2),B(I,3): NEXT
180 REM *** MENU
190 HOME
200 VTAB 3: HTAB 14: INVERSE :
PRINT "PATCHMENU" : NORMAL
210 VTAB 8: HTAB 3: PRINT "CHANGE
READBYTES(1)"
220 VTAB 10: HTAB 3: PRINT "ALTER
ERRORTRAPS(2)"
230 VTAB 12: HTAB 3: PRINT "CHANGE
WRITERBYTES(3)"
240 VTAB 14: HTAB 3: PRINT "EXIT
TOCOPYDISK(4)"
250 VTAB 16: HTAB 3: PRINT "QUIT
XXXXXXXXXXXX(5)"
260 VTAB 22: HTAB 6: PRINT "CHOICE
?(10-05)0" : GOSUB 1060:
PRINT
270 X = VAL (X$): IF X < 1 OR X >
5 THEN 260
280 IF X = 5 THEN END
290 IF X = 4 AND FL = 1 THEN CALL
8192: REM EXIT TO LS 6.0
FASTCOPY
300 IF X = 4 AND FL = 2 THEN POKE
103, PEEK (16379): POKE 104,
PEEK (16380): POKE 175, PEEK
(16381): POKE 176, PEEK
(16382): GOTO 80: REM EXIT TO
COPYA.REV
310 HOME : ON X GOSUB 340,610,750
320 GOTO 190
330 REM *** READ BYTE CHANGES
340 VTAB 3: HTAB 7: INVERSE :
PRINT "BYTEREADCHANGESFOR
RWTS" : NORMAL
350 VTAB 6: PRINT "Doyouwish
Addressbytechanges(Y/N) : :
GET Z$: PRINT
360 IF Z$ < > "Y" THEN 470
370 VTAB 22: PRINT "ENTERCHANGE
OR(N)FORONCHANGE"
380 VT = 9: FOR I = 1 TO 5
390 VTAB VT: PRINT "CHANGE"
;A$(I,1); "FROM";A$(I,2);
"TO" : GOSUB 1060: PRINT
400 CH$ = X$
410 IF CH$ = "N" AND I = 5 THEN
460
420 IF CH$ = "N" THEN VT = VT + 2:
NEXT
430 IF LEN (CH$) < 2 OR LEN (CH$)
> 2 THEN 390
440 GOSUB 1020:A$(I,2) = CH$
450 POKE A(I,FL),CH
460 VT = VT + 2: NEXT
470 VTAB 6: CALL Data058:CHANGES
"DoyouwishDatabytechanges
(Y/N) : : GET Z$: PRINT
480 IF Z$ < > "Y" THEN 590
490 VTAB 22: PRINT "ENTERCHANGE
OR(N)FORONCHANGE"
500 VT = 9: FOR I = 7 TO 11

```

```

510 VTAB VT: PRINT "CHANGE" ;A$
(I,1); "FROM";A$(I,2); "TO"
: GOSUB 1060: PRINT
520 CH$ = X$
530 IF CH$ = "N" AND I = 10 THEN
580
540 IF CH$ = "N" THEN VT = VT + 2:
NEXT
550 IF LEN (CH$) < 2 OR LEN (CH$)
> 2 THEN 510
560 GOSUB 1020:A$(I,2) = CH$
570 POKE A(I,FL),CH
580 VT = VT + 2: NEXT
590 RETURN
600 REM *** ERROR TRAP ALTERATIONS
610 VTAB 3: HTAB 6: INVERSE :
PRINT "ERRORTRAPCHANGESFOR
RWTS" : NORMAL
620 VTAB 6: HTAB 3: INVERSE :
PRINT "ADDRESSFIELDANDDATA
FIELDEDITS" : NORMAL
630 PRINT
640 FOR I = 1 TO 11
650 PRINT CHR$(I + 64); ".0"
;B$(I): NEXT
660 PRINT "L.RWTSChangesselect
ed-BacktoMenu"
670 VTAB 22: CALL - 868: INPUT
"SELECT LETTER&HITRETURN"
;Z$
680 IF Z$ < "A" OR Z$ > "L" THEN
670
690 Z = ASC (Z$) - 64
700 IF Z = 12 THEN RETURN
710 POKE B(Z,FL),B(Z,3): IF Z = 6
AND FL = 1 THEN POKE 9226,24
720 IF FL = 2 AND Z = 11 THEN POKE
47411,106
730 GOTO 670
740 REM *** WRITE BYTE ALTERATIONS
750 VTAB 3: HTAB 7: INVERSE :
PRINT "WRITEBYTECHANGESFOR
RWTS" : NORMAL
760 Address: byte"Doyouwish
Address: bytechanges(Y/N) : :
GET Z$: PRINT
770 IF Z$ < > "Y" THEN 880
780 VTAB 22: PRINT "ENTERCHANGE
OR(N)FORONCHANGE"
790 VT = 9: FOR I = 1 TO 6
800 VTAB VT: PRINT "CHANGE"
;A$(I,1); "FROM";A$(I,2);
"TO" : GOSUB 1060: PRINT
810 CH$ = X$
820 IF CH$ = "N" AND I = 6 THEN
870
830 IF CH$ = "N" THEN VT = VT + 2:
NEXT
840 IF LEN (CH$) < 2 OR LEN (CH$)
> 2 THEN 800
850 GOSUB 1020:A$(I,2) = CH$
860 POKE A(I,FL + 2),CH
870 VT = VT + 2: NEXT
880 VTAB 6: CALL - 958: PRINT
"DoyouwishDatabytechanges
(Y/N) : : GET Z$: PRINT
890 IF Z$ < > "Y" THEN 1000
900 VTAB 22: PRINT "ENTERCHANGE
OR(N)FORONCHANGE"
910 VT = 9: FOR I = 7 TO 12
920 VTAB VT: PRINT "CHANGE"
;A$(I,1); "FROM";A$(I,2);
"TO" : GOSUB 1060: PRINT
930 CH$ = X$
940 IF CH$ = "N" AND I = 12 THEN
990
950 IF CH$ = "N" THEN VT = VT + 2:
NEXT
960 IF LEN (CH$) < 2 OR LEN (CH$)
> 2 THEN 920
970 GOSUB 1020:A$(I,2) = CH$
980 POKE A(I,FL + 2),CH
990 VT = VT + 2: NEXT
1000 RETURN
1010 REM HEX-DEC CONVERSION
1020 CH = 0:L = 0:L = ASC (CH$):L
= L - 48 - (L > 64) * 7:CH = L
* 16
1030 L = ASC ( RIGHTS (CH$,1)):L =
L - 48 - (L > 64) * 7:CH = CH +
L
1040 RETURN
1050 REM INPUT ROUTINE
1060 X$ = "" : Z$ = ""
1070 GET Z$: PRINT Z$;
1080 IF Z$ = CHR$(8) THEN Z$ = ""
: POKE 36, PEEK (36) + 1: GOTO
1070
1090 IF Z$ = CHR$(13) THEN X$ =
"" : RETURN
1100 X$ = Z$
1110 GET Z$: PRINT Z$;
1120 IF Z$ = CHR$(8) THEN Z$ = ""
:LE = LEN (X$): IF LE = 1 THEN
X$ = "" : CALL - 868: GOTO 1070
1130 IF LE > 1 THEN X$ = LEFT$(
X$,LE - 1):LE = 0: CALL - 868:
GOTO 1110
1140 IF Z$ = CHR$(13) THEN RETURN
1150 X$ = X$ + Z$: GOTO 1110
1160 DATA ADDRESSPROLOGUEBYTE
#1,DE,13519,47505,13531,48302
,222
1170 DATA ADDRESSPROLOGUEBYTE
#2,AA,13520,47515,13532,48307
,170
1180 DATA ADDRESSPROLOGUEBYTE
#3,EB,13533,48312,235: REM W
BYTES ONLY
1190 DATA DATAPROLOGUEBYTE
#1,D5,13522,47335,13534,47187
,213: REM R/W BYTES
1200 DATA DATAPROLOGUEBYTE
#2,AA,13523,47345,13535,47192
,170
1210 DATA DATAPROLOGUEBYTE
#3,AD,13524,47356,13536,47197
,173
1220 DATA DATAEPILOGUEBYTE
#1,DE,13525,47413,13537,47262
,222
1230 DATA DATAEPILOGUEBYTE
#2,AA,13526,47423,13538,47267
,170
1240 DATA DATAEPILOGUEBYTE
#3,EB,13539,47272,235: REM W
BYTES ONLY
1250 DATA ADDRESSPROLOGUEBYTE
#1,9043,47447,0: REM ERROR
ADDRESSES & PATCH
1260 DATA ADDRESSPROLOGUEBYTE
#2,9054,47457,0
1270 DATA ADDRESSPROLOGUEBYTE
#3,9065,47468,0
1280 DATA ADDRESSPROLOGUEBYTE
#1,9095,47498,0
1290 DATA ADDRESSPROLOGUEBYTE
#2,9095,47498,0
1300 DATA ADDRESSPROLOGUEBYTE
#3,9095,47498,0
1310 DATA ADDRESSCHECKSUM,9095,
47498,0
1320 DATA ADDRESSEPILOGUES,9100,
47503,14
1330 DATA READERRORFLAG,9117,
47426,24
1340 DATA DATAPROLOGUEBYTE
#1,9134,47337,0
1350 DATA DATAPROLOGUEBYTE
#2,9144,47347,0
1360 DATA DATAPROLOGUEBYTE
#3,9155,47358,0
1370 DATA ADDRESSCHECKSUM,9203,
47406,0
1380 DATA DATAEPILOGUES,9208,
47411,14

```

Checksums

10-\$BADD	470-\$C3AC	930-\$A81C
20-\$9B13	480-\$214B	940-\$1D4B
30-\$4D3B	490-\$72D1	950-\$D7B3
40-\$AD92	500-\$0F8A	960-\$EA21
50-\$C899	510-\$DD39	970-\$30A7
60-\$DA01	520-\$1029	980-\$1DA6
70-\$DD06	530-\$D651	990-\$9D64
80-\$04F4	540-\$0433	1000-\$0ADE
90-\$16C4	550-\$2C9E	1010-\$8BD9
100-\$84CC	560-\$A9FC	1020-\$9C50
110-\$CEE7	570-\$E9FF	1030-\$19EA
120-\$592D	580-\$156C	1040-\$71DD
130-\$FDF4	590-\$F878	1050-\$F1EB
140-\$A835	600-\$8AF6	1060-\$BA8E
150-\$2BAF	610-\$A3FB	1070-\$661E
160-\$3F14	620-\$F171	1080-\$1962
170-\$88F1	630-\$5ADF	1090-\$39A9
180-\$17A1	640-\$B3C3	1100-\$04D5
190-\$F261	650-\$0983	1110-\$1A71
200-\$CB9D	660-\$EE51	1120-\$9516
210-\$2FB9	670-\$C6C8	1130-\$736D
220-\$767A	680-\$BFC4	1140-\$9A7F
230-\$0490	690-\$1FF4	1150-\$E7C4
240-\$26F1	700-\$9813	1160-\$FA6E
250-\$CDA9	710-\$8D8F	1170-\$1CA0
260-\$A507	720-\$72BF	1180-\$B5A9
270-\$9E46	730-\$578D	1190-\$2064
280-\$D815	740-\$2670	1200-\$D06F
290-\$63A1	750-\$45D0	1210-\$6804
300-\$B06E	760-\$49C3	1220-\$9662
310-\$5B11	770-\$3176	1230-\$077A
320-\$9E0D	780-\$558B	1240-\$013A
330-\$5195	790-\$867D	1250-\$48EA
340-\$8DAD	800-\$C73F	1260-\$FCEE
350-\$5B61	810-\$3A6D	1270-\$D722
360-\$7906	820-\$21A5	1280-\$FF9A
370-\$4CA4	830-\$BF25	1290-\$754D
380-\$4B90	840-\$0657	1300-\$2448
390-\$17C1	850-\$9E6B	1310-\$C292
400-\$A722	860-\$D1D4	1320-\$4652
410-\$5A7D	870-\$8710	1330-\$B752
420-\$F928	880-\$E317	1340-\$20FD
430-\$1801	890-\$8C37	1350-\$FE0A
440-\$CD02	900-\$B9B6	1360-\$EBAB
450-\$6DC1	910-\$AB09	1370-\$9C46
460-\$CB65	920-\$311E	1380-\$6DD1

Listing 3 - COPYA.REV

```

Load COPYA and add the following lines:
65 IF PEEK (16383) < > 2 THEN
PRINT CHR$(4);"RUN
LOADER.S6FCA"
72 POKE 16379, PEEK (103): POKE
16380, PEEK (104): POKE 16381,
PEEK (175): POKE 16382, PEEK
(176)
76 POKE 104,64: POKE 16384,0:
PRINT CHR$(4);"RUN SUPER 6.0
FASTCOPYA"
Run LOADER.S6FCA and it will allow you
to specify which copy program you are using. I
s.0 fastcopy or COPYA. This special loader
program will then either load Locksmith or
COPYA.Rev and then will run Super 6.0 Fast-
copyA. Note that if the COPYA option is taken,
COPYA is run briefly, just to load itself and
the Copy.IOB0 file, but exits to run Super 6.0 Fast-

```

copyA. (Note lines 72 & 76 in Listing 3). COPYA and its companion binary file remain low in memory while Super 6.0 FastcopyA is running at \$4001. I found this round-about method necessary as Super 6.0 Fastcopy changes RWTS in DOS 3.3 enough to prevent loading of the COPYA files from a normally formatted disk. This approach enables one to switch to COPYA at the appropriate time, without having to read the files from disk (See line 300 in listing 2).

When Super 6.0 FastcopyA is finally accessed, you will be allowed to edit RWTS painlessly prior to burning LS 6.0 or COPYA. By the way, if you do not as yet have a copy of LS 6.0 Fastcopy, run, don't walk for a copy of COMPUTIST v.43!

The Applesoft program, SUPER 6.0 FASTCOPYA, needs no extensive documentation. The DATA statements, lines 1160 to 1270 each contain 7 pieces of information regarding edits to change prologue and epilogue bytes so as to be able to read or write protected disks. In order, each line shows the name of the marker, its hex value, its initial read address in Locksmith (in decimal) and then in RWTS, its initial write address in Locksmith and then in RWTS, and finally its present value (in decimal). Lines 330-590 contain the routines for allowing one to change these bytes. Note that one enters changes in hex. These are converted to decimal values for poking into Locksmith or RWTS (for COPYA) by a simple hex-decimal conversion subroutine in lines 1020-1030.

DATA statements, lines 1280 to 1380 contain four items of information regarding pokes to abort the error traps. In each line, the error trap is named, its decimal change location is given, first in Locksmith and then in RWTS, and the byte to be patched is finally given. Lines 600-730 contain the routine to allow one to choose which error traps should be arrested. I have closely followed Bill Jetzer's approach here as it is simple and requires only one byte modifications. For those interested in comparing the Locksmith RWTS to DOS 3.3, do compare the READ routine for the address field at \$2344-\$239C in LS 6.0 Fastcopy to \$B944-\$B99F in DOS 3.3. The data field routine may be matched at \$239D-\$240B and \$B8DC-\$B943. Do look for the lookup table in page \$34. See the read bytes from \$34CC-\$34D6.

For writing to disk, I have also included the ability to patch the RWTS of Locksmith or DOS 3.3 for COPYA, so one can copy a normally formatted disk to produce a protected disk. See lines 740-1000 in listing 2 for this procedure. Note that the addresses required are read in the data statements in lines 1160-1270. For comparison, one can review the Locksmith write routine from \$2501-\$26D7 to normal DOS 3.3 RWTS from \$B82A-\$B8C1 and from \$BC57-\$BCC3. In further research into the art of protecting disks, there are some constraints. I have discovered that TS00, S\$00 must be of almost normal format. Boot 0, contained in ROM at \$C600 requires that the first sector read must have normal prologues, specifically D5 AA 96 for the address and D5 AA AD for the data field. Epilogue bytes are not checked and can be any values (normally DE AA). Therefore, one cannot change prologue bytes for TS00, but can modify the epilogue bytes. For further discussion of this point see "Fun with Super 6.0 FastcopyA" following this article.

After running LOADER.S6FCA and installing patches into Locksmith 6.0 Fastcopy or COPYA, the requested copy program is started. Note in line 290 that I expect Locksmith to start at \$2000 (dec. 8192). I have noted that some readers have experienced difficulties in obtaining a version of Fastcopy, and that some modifications require a start at \$2002. If this is relevant to your copy, change the CALL 8192 to CALL 8194 in line 290. I am not sure if my patches will work with all versions of LS 6.0 Fastcopy. If you have a different version, I would suggest scanning your version, comparing it to DOS 3.3 RWTS, and changing the data table in Super 6.0 FastcopyA accordingly. Please let COMPUTIST readers know of your work by writing RDEX.

Using Super 6.0 FastcopyA upon a few protected disks, I have found that the program works. I have not tried all variations yet, however. I did find that Locksmith seemed to behave a bit unusually, particularly when the read error flag was turned off. Locksmith seems to pause for a time when it encounters a bad sector, but then continues and seems to do a fine job. Pressing "V" prior to copying to access the validate function seems to prolong this pause. I find also that Locksmith does not always mark the unreadable sector, particularly if the read error flag is aborted.

Softkey for...

Ghostbusters

Activision

For a tutorial I present two step by step recipes demonstrating how one can use Super 6.0 FastcopyA. I have selected Ghostbusters as it has been previously softkeyed (COMPUTIST v.40, pp.7-8) by COPYA and readers may make comparisons. Ghostbusters has a simple protection. The second address epilogue byte (normally AA) has been modified on 3 tracks, to \$AB on TS01, to \$BC on TS12 and to \$CD on TS13. In TS00, S\$0A these trailer bytes are checked, but a

simple 6 byte edit will ensure a run-able program after these tracks are normalized.

Softkey 1

1. Place your DOS 3.3 disk containing LOADER.S6FCA, SUPER 6.0 FASTCOPYA and LS 6.0 FASTCOPY and run LOADER.S6FCA.
2. Answer "L" when requested for copy program.
3. Answer "2" for error trap changes.
4. Answer "E" when asked to make error trap changes (This will turn off the trap for address epilogue bytes).
5. Answer "L" (This will exit and return to the menu).
6. Answer "4" to start LS 6.0 Fastcopy
7. Place Ghostbusters disk in drive 1 and a blank disk in drive 2, and press return.
8. After the copy is made, sector edit TS00 S\$0A:

Trk	Sc	Byte(s)	From	To
\$00	\$0A	\$52	A5 04	A9 BC
		\$59	A5 04	A9 CD
		\$AA	A5 04	A9 AB

Softkey 2

As a variation of the method used in softkey 1, which turns off error traps, let us try another approach. Let us modify the 2nd address epilogue read byte for tracks \$01, \$12 and \$13. As Ghostbuster has these 3 tracks with different read bytes as well as the other 32 in normal format, one will have to run Super 6.0 FastcopyA 4 times, first to copy the normal tracks and then to copy the individual 3 protected tracks. Finally the sector edit must be done.

1. JRUN LOADER.S6FCA.
2. Ask for Locksmith by answering "L".
3. Press "4" to call LS 6.0 Fastcopy.
4. Place Ghostbusters in drive 1 and blank disk in drive 2 and press return (Note TS01, TS12 and TS13 won't copy, but all other tracks will).
5. Boot up DOS 3.3 again and RUN LOADER.S6FCA.
6. Ask for Locksmith and answer "1" to change read bytes.
7. Answer "Y" when asked if you wish to change Address bytes.
8. Answer "N" to the following 4 prompts (We are not interested in making prologue or 1st epilogue byte changes).
9. Enter "AB" to the Address epilogue byte #2 prompt.
10. Answer "N" when asked if you wish to change Data bytes.
11. Press "4" to call LS 6.0 Fastcopy.
12. Upon seeing Locksmith, enter the following:
0008:01 Set beginning track
0009:01 Set ending track
return Start Locksmith to copy track \$01, ignore the write asterisk as it really does copy!
13. Repeat steps #5 to 12, but enter "BC" in step #9 and set beginning and ending track to 12 in step #12 (0008:12 and 0009:13).
14. Repeat steps #5 to 12, but enter "CD" in step #9 and set beginning and ending tracks to 13 in step #12 (0008:13 and 0009:13).
15. Make the same sector edits in TS00, S\$0A as in softkey 1.

The reader will note that softkey 2 is much more involved than softkey 1. I have presented this latter softkey to demonstrate a minor problem with LS 6.0 Fastcopy. If Locksmith cannot read a track it will still format and write the track (with zero bytes if no information has been read). If it has read valid information, but the read bytes are different from the write bytes, it will write the information in the buffer, but will display an asterisk, signifying no write. I found, however, that the write operation had been accomplished without error!

Fortunately, LS 6.0 Fastcopy provides a method of not recording information on tracks previously copied correctly. After copying the normal tracks, when protected tracks are to be copied (after patches have been done with SUPER 6.0 FASTCOPYA), one can specify the tracks to be copied by entering a simple code into the sector copy program prior to starting the copy function. The starting track can be given by entering [0008:xx], where xx is the starting track in hex. Then the ending track may be given by entering [0009:yy], where yy is the ending track. LS 6.0 will then read and write the appropriate track(s) and not overwrite previously written tracks.

The reader will, no doubt, see where either the softkey 1 or the softkey 2 approach might be appropriate. Using error trap changes will fit those simple situations where only a few byte changes have been made in a protected disk. Thus one would answer "2" in the menu of SUPER 6.0 FASTCOPYA and then make appropriate error trap changes. But if the address and data fields are extensively modified, one must follow the softkey 2 approach. I can see the need for softkey 2 for the great number of protected disks which have a normal track \$00 and protected tracks \$01-\$22. For disks having changes from track to track and sector to sector, if softkey

1 won't work, then Super IOB and the design of a special controller would be more appropriate.

Fun with Super 6.0 FastcopyA (and a sector editor)

An unusual capability of Super 6.0 FastcopyA is to copy a disk to a non standard format. This would allow one to produce a copy protected disk that would prevent others from obtaining copies of your private programs. This protection is simple, involving only changed address and data field marker bytes. It would necessitate using a bit copier, however. Here are two examples.

Copy Protection method 1

Using Super 6.0 FastcopyA, we have the capability of altering any combination of prologue and epilogue bytes for writing a disk during the copy process. One must bear in mind, however, that TS00, sectors \$00-\$09 generally must have normal prologue bytes. This is because the boot 0 in ROM (\$C600-\$C6FF) expects normal prologues in S\$00 of track \$00. Unless this sector, when read into \$B600, has extensive modifications, the next 9 sectors also must have normal prologues. These sectors contain DOS 3.3 RWTS and are then used to load other tracks and S\$0A-\$0F on TS00. These can have altered prologue markers. Our difficulty comes from the fact that LS 6.0 cannot copy different markers to different sectors on any one track. Since track \$00 requires 2 different types of prologue markers for a protected disk, one can not use LS 6.0 for this type of protection directly. There is a way, of course. See protection method 3 for details.

This first method of copy protection shows how to produce a very simple result: a disk with modified epilogue bytes. Follow these simple steps:

1. JRUN LOADER.S6FCA and choose Locksmith ("L").
2. Enter "3" in the menu to choose "Change Write Bytes".
3. Answer "Y" to the request for changes to the address field.
4. Change the 2nd Epilogue byte from AA to AB. Change the 3rd Epilogue byte from EB to EE. (Answer "N" to everything else in the address change field)
5. Answer "Y" to the request for changes to the data field.
6. Change the 2nd Epilogue byte from AA to AB. Change the 3rd Epilogue byte from EB to EE. (Answer "N" to all else.)
7. In the menu ask for "4" to obtain LS 6.0 Fastcopy.
8. Insert a normal disk in drive 1 and a blank disk in drive 2. Start the copy by pressing return.
9. Using a sector editor, make the following edits on the copy:

Trk	Sc	Byte(s)	From	To
\$00	\$02	A3	AA	AB
		AB	EB	EE
\$00	\$03	3F	AA	AB
		9B	AA	AB
\$00	\$06	B3	AA	AB
		B8	EB	EE

10. You now have a protected copy, but do not boot this disk and try to write to a normal unprotected disk. That would cause even the saintliest of individuals to develop an instant four-letter word vocabulary!

Copy Protection method 2

Either follow steps #1 to #9 above or use your protected copy to produce a more highly advanced protected disk. (One can also use Super 6.0 FastcopyA to obtain a copy by changing both the read bytes and the write bytes).

In COMPUTIST v.39, p.18, Rich Etarip (piratE) presented a delightful approach to producing a disk that would easily copy, but the copy would not boot. His approach was to patch a subroutine into DOS to check for the 3rd epilogue byte (normally EB). Copiers do not check for this byte and simply use the normal EB byte when writing. The copy will ask for the changed byte upon booting, but lo and behold, it's not there! Note that in method 1 that I have purposely changed this 3rd epilogue byte to EE from EB, especially so we can try this method. Let us plagiarize Rich's approach and sector edit the boot 1 routine on TS00, S\$00 (This routine is at the start of DOS 3.3 RWTS from \$B600 to \$B6FF).

xx4A: 4C B4 B6
xxB4: 20 C0 B6 4C 00 B7
xxC0: A2 60 BD 89 C0 BD 8E C0
xxC8: A9 00 20 A8 FC A9 00 20
xxD0: A0 B9 A0 04 BD 8C C0 10
xxD8: FB C9 DE D0 F7 BD 8C C0
xxE0: 10 FB C9 AB D0 EE BD 8C
xxE8: C0 10 FB C9 EE F0 06 88
xxD0 E2 4C F2 B6 BD 88 C0 60

Note the location of the 3 trailer bytes at \$DA, \$E3 and \$EC. These can be altered to suit the occasion! After the above sector edits have been made, a protected disk with 2 levels of protection has been accomplished. The ingenious COMPUTIST reader may make things much more difficult by either checking for the existence of this routine on disk from some of the programs, or calling the disk check several times (CALL

\$46784 from Applesoft or JSR \$B6C0 from assembly).

Protection method 3

So far we have produced a protected disk with altered epilogue bytes and a check for the 3rd byte for an extra bit of fun. Now let us start with changing prologue bytes! Here we may have a bit of a problem. Remember that Boot 0 loads TS00, S\$00 and expects normal prologues, but does not check for epilogues. Boot 1 loads in S\$01 to S\$09 with normal prologues and possibly changed epilogues (yes, we already did this!). But then Boot 2 (at \$B700 or S\$01) uses RWTS which is now loaded in. If the prologue bytes are changed in RWTS, then TS00, S\$0A to \$0F will have altered prologues. How do we copy S\$00-\$09 with one set of prologues and S\$0A-\$0F with another set with LS 6.0? The answer is, we can't. However, if we use a sector editor, such as CIA or equivalent, perhaps we can. Here is the procedure I followed:

1. Have ready the disk developed by protection method 1 (the source disk) and a blank disk (the destination disk).
2. JRUN LOADER.S6FCA
3. Get the Locksmith option by answering "L".
4. Select "1" to select read byte changes
5. Change the 2nd epilogue byte in both the address and data fields from AA to AB
6. Select "3" from the menu to select write byte changes
7. Change the 2nd epilogue byte in both fields from AA to AB. Change the 3rd epilogue byte in both fields from EB to EE
8. Select "4" in the menu to copy
9. When LS 6.0 Fastcopy appears, enter 0008=00 and 0009=00 to read and write TS00 and press return to copy TS00 (original source disk in drive #1, blank destination disk in drive #2).
10. Repeat steps 1 to 6 above
11. Change the Epilogue bytes in both fields as in step 7. Also change the 1st data prologue byte from D5 to 97.
12. Select "4" again and when Locksmith appears, enter 0008=01 and 0009=22 to copy TS01-\$22. (Do keep track of where your disks are at all times - if you are not sure what you are doing from my abbreviated instructions, write protect disks that you do not wish to be destroyed!)
13. Additional sector edits should now be done to the copy (destination) disk to include the changed data prologue byte (D5 has been changed to 97). Using CIA one can set both read bytes and data field write bytes. Set both read epilogue fields to DE AB. Set the write data field to D5 AA AD and DE AB EE (Copy II+ also has this capability). Make the following sector edits:

Trk	Sc	Byte(s)	From	To
\$00	\$02	53	D5	97
		E7	D5	97

14. Keeping the settings in #13, make one additional setting: Change the 1st data prologue write byte to 97 from D5. Now read one by one, track \$00, sectors \$0A to \$0F and write them back to the copy disk. This will only alter the initial data prologue byte so RWTS can recognize and read these sectors.

Now that you are finished, boot up your protected disk and enjoy. Write an article to COMPUTIST on methods to deprotect it!

B. Walter Disney MD

Softkey for... Children's Writing & Publishing Center

The Learning Company

Requirements:
COPYA
2 Blank disks

Let's begin with a quick review. This is a desktop publishing program for children (and those young at heart). It uses mouse or keyboard commands. My seven year old and I found using this program extremely easy. The graphic output is truly outstanding, even with a black and white printer.

The operating screen uses icons with a label, that enabled my son to use the program solo. There are eight on screen fonts to use. The program will not let you place graphics and text incorrectly, it prompts you as to the programs restrictions. What you see is what you get. This program is simple and fun to use.

The program has two disks. We need only concern ourselves with side one of the program disk. The other three sides are COPYA-able.

1. Load and run COPYA
- 70
- CALL-151 enter monitor
- B942: 18 ignore epilogue errors
- 3A1: 18 ignore any DOS errors
- B956: 00 ignore data address checksums
- 3D0G to start COPYA

That's it folks. You have now ignored the bad sector (track 12 sector 0). Enjoy!

George Sabeh PA

Softkey for...

Arkanoid

Taito

The disk is in normal format and can be copied by COPYA. As can be expected, the copy will not boot. I was able to study a cracked copy by the Necromancer and was able to determine the protection. Credit for this should go to the Necromancer.

1. Copy the original using COPYA or Locksmith Fastcopy.
2. Boot your favorite sector editor and perform the following sector edit:

Trk	Sct	Byte(s)	From	To
20	00	13-14	D0 03	EA EA

If unable to find the above, then search for AD F8 40 D0 03 4C 00 02.

Softkey for...

Dark Lord

Datasoft

This program will copy with COPYA but produces errors on track 6, both front and back sides. I was able to discover the protection by studying a cracked copy. Credit for this softkey should go to the Sheik.

1. Copy both sides with a copy program that will ignore errors. I used Locksmith Fastcopy.
2. Boot your favorite sector editor and perform the following sector edit:

Trk	Sct	Byte(s)	From	To
3	5	0-2	FC D9 F0	A8 D0 CD

Softkey for...

Platoon

Data East

This program will copy with COPYA, but the copy will not boot as expected. The copy will reboot immediately after trying to boot it. On a hunch, I scanned for 4C 00 C6. This is what tells the disk to reboot and for some of the beginners it is the same as typing C600G from the monitor or PR#6 from BASIC to boot the disk. I was lucky and found it on the first try and was able to disable this and produce a deprotected copy.

1. Copy original with any whole disk copier such as COPYA.
2. Sector edit:

Trk	Sct	Byte(s)	From	To
0	E	4A-4C	4C 00 C6	EA EA EA

Softkey for...

Ice Demons

Morningstar

This program uses an abnormal fast DOS with a text file as the boot program.

1. Boot DOS 3.3 or preferably a fast DOS such as ProntoDOS.
2. Type the following as your HELLO program and initialize the disk:

```
NEW
10 PRINT CHR$(4)"EXEC RUND0"
INIT HELLO
```

Note that a text file named RUND0 is the boot program the disk uses.

3. Boot Ice Demons and break out into the monitor. This should be done by pressing ctrl reset shortly after the program starts booting. If you wait too long it will not work. Verify that you have the RWTS in place by checking for valid data at B800. I achieved the same thing by using the NMI from the Senior PROM, but this is not necessary.
4. Move the RWTS to a safe location such \$1900. 1900<B800.BFFF
5. Boot a slave disk and enter the monitor. Then save the RWTS to your Super IOB disk. Then boot your Super IOB disk and run the Swap Controller and copy the disk to a working copy. Follow the instructions with the Super IOB. This disk now can be cataloged but will not boot.
6. Finally, copy all the files from the working copy to the initialized disk created above.
7. Alternatively you can use COPYB to make a working copy and note that it requires the RWTS at 8000 for COPYB to work properly.

Softkey for...

Sierra Programs:

Sierra OnLine

I found that Sierra uses several layers of protection. If you find that the program is not running correctly during the game, you should look for further protection. I have run into this on several of their releases. I have compiled the following list of protection schemes that I am aware of but they may not be the only ones they use.

1. Scan disk for A9 00 AA A8 and change to 60 00 AA A8.
2. Scan disk for 20 00 FF and change to EA EA EA. This is usually found in several locations.

3. Scan disk for C9 C9 D0 0D and change to C9 C9 29 00 or 29 00 D0 0D.

Advanced Playing Technique for...

Xevious

?

You can prevent decrementing the number of lives you have and thus achieve unlimited lives by searching for CE 2B 60 and change this to EA EA EA. I found this on track 17, sector 0, byte 18-20.

Advanced Playing Technique for...

Victory Road:

?

You can delete most of the monsters and thus be able to see all the program to its end by sector editing track 19, sector 1, byte 17-18 and change from E9 01 to EA EA.

Keith B. Reed CA

Removing the Manual Check from Pirates! GS

Requirements:

Apple IIgs 256K
Deprotected Pirates GS

After deprotecting Pirates GS with Computist #61, pg. 8, by Dan Halfwit, I still had a pet peeve about the manual check. There is a simple way to bypass the manual check routine by placing a RETURN in line 1121 just after "Y = 10:" in the basic file called PICK. I put a REM after the RETURN so that the ending of the original line would remain intact. This is the way the line should look:

```
1121 IF ZZ = 41 THEN X = 4 : Y =
10 : RETURN : REM This RETURN
bypasses the required Doc's :
GOSUB 8200 : POKE PR,C+C : GOTO
1130
```

Everything in this line after the required RETURN can be left out. A previous attempt to elevate this manual check bypassed part of the program.

For those that are not sure how to go about making this change on their disk, here is the procedure:

1. Make a copy of your deprotected Pirates GS disk. Load BASIC System and insert the deprotected copy of the Pirates GS disk.

LOAD PICK

LIST 1121

1121 IF ZZ = 41 THEN X = 4 : Y = 10 : RETURN

LIST 1121

SAVE PICK

is this the correct line?

check for typo's

Enjoy the added convenience of not looking for your Pirates manual.

Daniel Bashford Australia

Playing Tip for...

Who Framed Roger Rabbit

Buena Vista Software

The following questions have to be answered after the completion of each level, one question is asked, here are the answers. I must thank The Apple Odessa and The Buzzard for their help.

The answers are in capitals and the numbers are spaced out.

Marvins Sales Manager is MR SMITH
 A bottle of sound effects costs 496 cents
 A bottle of earthquake pills costs 29 cents
 Stay Put glue costs 69 cents
 Each Squirting flower costs 19 cents
 Vanishing creme comes in a 70 ounce jar
 FOOTPRINTS come with a Dancing booklet
 HOT GUM costs one cent
 Itching powder costs 72 cents
 There is 99 pinches in a jar of itching powder
 Mighty smelly Limburger cheese comes in a 12 ounce WHEEL

Alum comes in a one quarter CARTON
 Magic lamps are for anyone who is in a PICKLE
 Magic Lamps are made of BRASS
 It takes 3 DAYS to recharge a seltzer bottle
 Shrinking potion comes in a 3 ounce and a 9 ounce bottle
 A bear trap costs 99 or 990? cents
 Rocket Roller skates can break the SOUND BARRIER

Backward pills make your victim WRITE BACKWARDS

Riot 7a 16ton weights will DRIVE your victim into the ground

A 16ton weight can SHATTER your victim into a thousand pieces

The 16 ton weight does not come with rope, etc or VICTIM

A 16ton weight can turn your victim into a MANHOLE COVER

You can use the sound effects to turn your POPGUN into big berth

Sound effects come in corked up LEAKPROOF bottles

If you order all 4 sound effects bottles Marvin will throw in a free BRONX cheer

Earthquake come from the planet SHIVER-BERRY

In California Shiverberry plants are grown in SAN ANDREAS

Stay put glue can get you out of STICKY situations

Squirting flowers come in carnation, orchid and ROSE

The Pogo stick is powered by PLUTONIUM

The giant slingshot is made of OAK

You can throw a cream pie 50 yards

The vacuum is a SUCK-O-LUX brand

Vanishing creme lasts until your next BATH

When someone sits on a whoopee cushion they may blame the DOG

You can always impress the GIRLS with hot gum

Elevator shoes can turn puny runts into towering BEHMOTHS

Elevator shoes come in 2 and 4 floor models

The box o mallet can reach 50 FEET

A good place to put an exploding cigar is in your foes PUSS

The giant slingshot can launch a small ELEPHANT

The hand buzzer is still our biggest SELLER

Itching powder lasts about 6 MONTHS

Mighty smelly linburger cheese is aged over 90 years

Alum works better than LEMMONS

You get a free bag of WATCHES with the magic lamp

Magic lamps are sold in AS-IS condition

Seltzer can shoot up to 250 feet

Shrinking potion was developed by a MONROVIAN headhunter

A bear trap comes complete with ANCHOR spike and 2foot chain

Rocket roller skates are powered by NITRO-GLYCERIN

X-Ray glasses are made with VISULONIUM

X-Ray glasses are not effective on LEAD

Curve balls are made by angry AUSTRALIANS

The giant magnet can pull flying SAUCERS out of the sky

Perma Slip banana skins remain slippery for DAYS

Fine print comes from the offices of Dewey, CHEDAM and Howe

The man eating plant is also known as the di-onadea muscipula GIGANTICUS

A man eating plant can grow up to 23 feet tall

The magic carpet flies like a DREAM

The year of the gag factory catalog is 1947

To order from the catalog simply dial Klondike 512

The last direction on the that-a-way sign in AZUSA

Portable holes have a ROOMY interior

Man eating plants are grown only in CUCAMONGA

Springy eyeballs come in blu, brown, hazel or BLOODSHOT

Thud! Wap! Pow are all SLAPSTICK sound effects

Gurgle, Woosh, Splash are WATERWORKS sound effects

Whirr, Buzz, Ta-Pocketa are all CONTRAPTION sound effects

Bang! Kapow PShhht are all EXPLOSION sound effects

Earthquake pills come 50 to the bottle

The pogo stik has EZ-grip handlebars

Stay put works on toons of even incredible STRENGTH

The giant slingshot has a one hundred percent INDIAN rubber band

Eat HOT GUM and flames will come out of your mouth

The box o matic mallet looks like a standard issue ANTI-MOUSE

ELEVATOR shoes are always good for a lift

This extra stinky, extra sticky cheese is the most potent that marvin has ever PEDDLED

The magic lamp comes with a PAPER bag filled with watches

The magic carpet is gaurenteed for twelve thousand miles or 12 washings

Magic carpet instructions come on a PARCHMENT scroll

Cream pies come with an exclusive FACE-SEEKING feature

Who Framed Roger Rabbit? JUDGE DOOM

Marvins gag factory is on CENTRAL AVENUE

BACKWARD PILLS cost the same when they are on sale

Marvin will be happy to sell you a sixteen ton WEIGHT

Drinks are served in the ink and pen club by PENGUINS

You get 1000 sheets of fine print for your hard earned seventy nine cents

38 items are available in the gag catalog

EARTHQUAKE pills are gaurenteed to shake things up

ALUM will make your buddy pucker till the cows come home

You can soak some sucker at a SHINDIG with the seltzer bottle

The MAGIC CARPET is hand embroidered in Bagdad

The EXPLODING CIGAR is on the front cover of the gag catalog

HOT GUM is the least expensive gag

siht ekil skool taht noitpirced a sah gag tahW (What gag has a description that looks like this?) BACKWARDS PILLS

Who wrote the mumbo jumbo for the fine print? JUDGE DOOM

How many things can you catch with the bear trap? 36

What gag costs a lot of semolinas? MAGIC LAMP

Thats it, perhaps I will simply figure out a sector edit for next issue or perhaps someone else could, also a patch for unlimited lives would be good as it takes a while to reload each time.

The actual file that contains all these is on DISK 1, side 2, it is a ProDOS disk, the file is a binary one called TEST, you can print it out using Copy II Plus, or convert it to a text file.

Does anyone have APTs for Rampage or Gauntlet?

Advanced Playing Technique for...

The Bards Tale III

Electronic Arts

Thanx to AGENT STEALTH of [COD]

Character Storage

Track 22, sectors 00-0F on the character disk.

Byte(s) description

00-0F Character's Name

10-14 Str, IQ, Dex, Con, Luck in respective order

15-18 Character's Experience Points

19-1C Character's Gold

1D-1E Character's Level

1F ??????????????????

20-21 Current Hit Points

22-23 Maximum Hit Points

24-25 Current Spell Points

26-27 Maximum Spell Points

29 Class (See Note:Class)

2A Race (See Note:Race)

2B Sex (00 - Male, 01 - Female)

2C Character Illustration(See Note:Illus.)

30-53 Weapons (See Weapon's List)

54-63 Spells (See Note:Spells)

64-66 Special Abilities(See Note:Spec. abil)

General Notes

For all bytes between (10-1E) and (20-27) the maximum value for each byte is "FF". For things that require 2 or more bytes (such as gold, experience, hitpoints etc.) the combined value of those 2 or more bytes will give you that number in decimal. For example:

Current Hit Points is stored at \$20-21. If the values there are 00 & FF then the total is \$00FF or 255 decimal. This applies to hit points and spell points.

In gold, experience and levels it's basically the same except you must combine in reverse order. If the values at \$15-18 are FF 00 00 00 then the total, in reverse, is \$000000FF or 255 decimal. It's quite simple.

Note:Class

Here are the hex values for the different classes in byte \$29. Note: bytes \$0D-OE should never be used!

00-Warrior	05-Rogue	0A-Arch Mage
01-Wizard	06-Bard	0B-Chronomancer
02-Sorcerer	07-Paladin	0C-Geomancer
03-Conjurer	08-Hunter	0D-Monster
04-Magician	09-Monk	0E-Illusion

Note:Race

Here are the hex values for the different races in byte \$2A.

00-Human	02-Dwarf	04-Half Elf	06-Gnome
01-Elf	03-Hobbit	05-Half Orc	

Note:Illus

These must be changed so you won't get a male fighter looking like a female Conjurer.

21-Male	30-Female	36-Male	37-Female
Fighter	Fighter	Magician	Magician
Bard	Bard	Conjurer	Conjurer
Rogue	Rogue	Sorcerer	Sorcerer
Paladin	Paladin	Wizard	Wizard
Hunter	Hunter	ArchMage	ArchMage
		Chronomancer	Chronomancer
		Geomancer	Geomancer

All the other values are used for other monsters.

Note:Special ability

These bytes are used in different ways. It all depends on the class. (What each byte does for each class)

Class	\$64	\$65	\$66
Bard	# of Songs Left (FF-max)	Types of Songs (FF-All Songs)	Nothing
Hunter	Critical hit % (FF-99%)	Nothing	Nothing
Rogue	"FF" for all 3 bytes will give you 99% in all Rogue Abilities!		

Note:Spells

Spells were a pisser to find. So if you want all the spells in all the bytes specified insert a FF. It seems they use a special system which would take hours to figure out.

Weapons Listing

Here are the weapons listing. Every 3 bytes will contain all the information for one weapon. For Example:

Byte 30- Weapon	00-Not Equipped
	01-Equipped
	02-Non-usable
Byte 31- Weapon Type (See List)	

Byte 32- Number of Charges (FF-Unlimited Supply!)

Byte 33- Weapon 00 01 02

Byte 34- Weapon Type
Byte 35- Number of Charges
etc, etc, etc.

Note: 12 equipment is max.

Weapons Listing

Table listing various weapons and their stats, including items like torch, sword, dagger, and armor.

7. Follow the above procedure for the Tutorial program.

That's it. You now have a deprotected copy of Perplexing Puzzles and the Tutorial.

Softkey for...

Freddy's Puzzling Adventures

Developmental Learning Materials (DLM)

This is a cute little math game for young children. But, like other games from DLM, the disk is copy protected. Here is how to make a deprotected copy of this program.

1. Boot a DOS 3.3 disk.
POKE 47426,24 Disable Chksum & Epilog Errors
RUN COPYA

2. Boot your favorite sector editor.

Table with columns: Trk, Scl, Bytes, From, To. Values: \$00 \$03 \$42 \$38 \$18

You now have a deprotected copy of the program.

Softkey for...

Pixelwerks/Mr. Pixel's Programming Paint Set

Mindscape/Master Software Inc

Mr Pixel's Programming Paint Set is a paint and drawing program for children. It is easy and interesting to use but it is copy protected, and trying to make a backup using EDD 4, Locksmith, or Copy II + did not work. However, here is a solution to the problem of making a backup copy.

1. Boot DOS 3.3 System Master disk.
POKE 47426,24 Disable chksum & epilog errors.
RUN COPYA

2. Boot your favorite sector editor and scan for BD 8C C0 10 FB C9 E7. I found it on track \$11, sector \$03. The first E7 should be followed by D0 2C. The second E7 should be followed by D0 23.

Table with columns: Trk, Scl, Bytes, From, To. Values: \$11 \$03 \$49 \$52 \$7C E7 E7 38 AA AD 18

Write the changes back to your copy and you are finished.

B.M.E. Upp ("Scotty") CT

Wow! What was it that someone said about good things coming in small packages? I refer to "Southern Exposure"'s crack for the mid-80's version of "Magic Spells" (Computist #67) - one of the smallest, easiest, most straight-forward cracks I've ever seen - and it works! This brought to a close my two-year search for a crack for this disk. Nothing else worked, not CopyII+, not the other softkeys I found in Computist, nothing! As far as I'm concerned "Southern Exposure" is "Southern Comfort", and my faith in Computist has been amply rewarded.

Now for my first "contribution" to the Exchange, albeit only an enhancement. I really liked D.B. Brett's "FIND.CAT" program (same issue), but didn't like being "locked in" to (either) printer or screen output only. As a general rule, I don't like being locked into, or out of for that matter, anything, which is one of the reasons that I'm a Computist!. Ergo, I added a few lines of code to provide for selection of output to screen OR printer. This was somewhat complicated by the fact that I wanted to retain the 80 column format for screen output, too - while not running the entire program under 80 columns. Thus, I had to arrange for the screen to flip back and forth between 80 and 40 columns. Naturally, your machine must have 80 column capability in order to utilize this enhancement.

Add or change the following lines:

PATCH FIND.CAT

200 TI = 0:N1 = 0:N2 = 0:N3 = 0:N4 = 0:OF = 35:PR\$ = "" :PR = 0
560 VTAB 5: HTAB 1
570 PRINT
"Send to (P) rinter (or) S) creen? 0"
; GET PR\$: IF PR\$ <> "P" AND
PR\$ <> "p" AND PR\$ <> "S" AND
PR\$ <> "s" THEN 560
575 IF PR\$ = "P" THEN PR = 1
580 IF PR THEN PRINT D\$ "PR#1" :
PRINT CHR\$(9); "80N" : GOTO
600
590 PRINT D\$ "PR#3"
910 IF N4 = 0 THEN PRINT
"NO FILES PRE SENT" : ON PR GOTO
970: GOTO 1160
970 IF PR THEN PRINT D\$ "PR\$0"
1170 PRINT : PRINT
"ANOTHER DISK? 0" ; GET Z\$:
PRINT : PRINT D\$ "PR#0"

The changes in 950, 970 and 1170 are there to prevent returning to 40 column mode prematurely - thus wrecking the screen display.

And now, a question: the "FIND.CAT" program seems to work only with disks that have at least one binary file on them; on disks without such a file, the drive makes terrible noises and the program claims a drive error! Missing Applesoft, Text, or Integer files do not produce this result.

Can someone tell me why this happens, and suggest a correction?

Thanks for the best "Hacker's Mag" of all!

Datablaster NV

Softkey for...

VCR Companion

Broderbund

This is my first contribution to Computist. VCR Companion uses ProDOS and will copy with a normal copier, giving no errors. However, the copy will not boot. This told me that some type of signature check routine was being used. After determining where the signature check program was (in the boot file VCRC.SYSTEM), and making the following edits, the copy worked with no further problems.

- 1. Make copy of VCR Companion using any normal copier, both disks, both sides.
2. Boot (load) ProDOS BASIC.
3. Insert the copy of VCR Companion, disk 1, side 1, into drive 1.
4. At the BASIC Prompt "J", type the following: BLOAD VCRC.SYSTEM, TSYS, A\$2000
POKE 8462,76
POKE 8463,46
POKE 8464,33
BSAVE VCRC.SYSTEM, A\$2000, L\$61F, TSYS

Dennis Kathrens TX

Softkey for...

Talking Stickybear Opposites

Optimum Resource

Requirements:
Apple IIgs 512K
3.5" disk copier
3.5" disk editor

Talking Stickybear Opposites is another educational program by Optimum Resource. I bought Talking Stickybear Alphabet and (what great timing!) in my next issue of Computist (#66, page 6) was the deprotect which allowed me to run it from my hard disk without the annoying "Master Disk" check, which allows the program to load from a copied diskette but demands that the original be placed in the drive before the program will continue.

I used the CopyII+ bit copier and the Stickybear Alphabet parm to make a copy of my original disk, and started working on that. Using Block Warden from the ProSel package, I tried searching for the bytes listed in that deprotect, but couldn't find them on my disk. The bytes were a command to JSR to the protection routine, which was listed in the softkey for Stickybear Alphabet.

Since I am not too good at code tracing yet, I didn't know how to determine the address for the entry point of the protection routine on the Talking Stickybear Opposites disk, which would be the address to JSR to for the protection routine. Hence the bytes for the Talking Alphabet Softkey were not applicable to my Talking Opposites disk.

I assumed that the protection routine would be much the same though, so I searched for the first six bytes of the Stickybear Alphabet protection routine on my Stickybear Opposites disk. Sure enough, there it was, differing only in the predictable places, like parm table addresses. I saw that to satisfy the protection routine, the carry flag would have to be set before returning to the subroutine from which the protection routine was called.

I did the quick and dirty thing. I went right to the start of the protection routine and changed the first byte to a Set Carry (SEC) opcode, the second byte to a Return to Subroutine (RTS) opcode, and the next two bytes to \$00 00 (BRK BRK). I then wrote the changed block back to my working copy and booted the disk. To my delight, it fooled the program and it loaded and ran fine!

This is the first time I have been able to successfully apply the knowledge I have gained from subscribing to and reading Computist, and I am quite proud of myself. Thanks to the unknown author of the Talking Alphabet softkey for the listing of the protection routine.

The start of the protection routine was found at relative block \$0106, byte \$163 (relative byte \$396E) and starts out with 22 A8 00 E1 20 00. To disable the routine, change this to 38 60 00 00 20 00.

Could someone write an article discussing the methods for determining where in memory a program or routine will be loaded from disk? This has always mystified me, even back on the good old IIE. Now, with the whiz-bang Memory Manager on the IIgs, and all the resulting relocatable code being written, I am even more confused.

Blain Johnson WY

Ultimapper V

I'm sure that more than once while playing Ultima V, you have traveled into the Underworld and gotten lost. Wouldn't a map be helpful? Well with some simple logic and a heap of luck, I

managed to make a mapper of the Ultima V realm (Britannia and Underworld). First of all, I created a BASIC program. BASIC, although a very versatile language, can be very slow at times, so I sat down and converted the BASIC program to Machine Language.

Type in the listings in DOS 3.3 and save them:

ULTIMAPPER V

0A00: 20 7E 0A 8D D0 D2 C9 CE \$E49A
0A08: D4 C5 D2 A0 D3 CC CF D4 \$EEAE
0A10: A0 A8 B0 AD B7 A9 BA 00 \$7C6D
0A18: A9 A0 20 1B FD C9 B8 B0 \$C0DE
0A20: F7 C9 B0 90 F3 F0 50 20 \$3F24
0A28: ED FD EA EA EA EA 8D 7A \$188B
0A30: 0A 20 8E FD 20 8E FD 20 \$1F5D
0A38: 7E 0A C9 CE D3 C5 D2 D4 \$5113
0A40: A0 D5 CC D4 C9 CD C1 A0 \$592D
0A48: D6 A0 A7 CD C1 D0 A7 A0 \$9F92

0A50: C4 C9 D3 CB A0 C9 CE A0 \$B612
0A58: C4 D2 C9 D6 C5 A0 B1 8D \$51D3
0A60: DB C5 D3 C3 DD A0 D4 CF \$725E
0A68: A0 C5 D8 C9 D4 00 A9 A0 \$E693
0A70: 20 1B FD C9 9B D0 04 60 \$ADD8
0A78: 00 00 B1 4C 60 0B 68 AA \$D80D
0A80: 68 A8 8C 8D 0A E8 D0 03 \$79AE
0A88: EE ED 0A 4C 00 0A F0 06 \$F5CC
0A90: 20 ED FD 4D 85 0A 98 48 \$6A77
0A98: 8A 48 60 0F 01 02 03 04 \$5721

0AA0: 05 06 07 08 09 0A 0B 0C \$F74D
0AA8: 0D 0E 00 A0 A7 A9 20 8D \$DE56
0AB0: F1 B7 98 48 8C C1 0A A9 \$D08E
0AB8: 13 8D EC B7 A0 03 98 48 \$A423
0AC0: B9 A7 0A 8D ED B7 A9 01 \$F9B6
0AC8: 8D EA B7 A9 01 8D F4 B7 \$A39B
0AD0: A9 00 8D EB B7 8D F0 B7 \$09B0
0AD8: 8D F3 B7 18 20 E3 03 20 \$45F1
0AE0: D9 03 B0 F7 EE F1 B7 68 \$303D
0AE8: A8 88 EA 10 D1 EE EC B7 \$3012

0AF0: AD EC B7 C9 23 D0 C5 A2 \$15DB
0AF8: 20 A0 24 8E 0D 0B 8E 2D \$648A
0B00: 0B 8C 26 0B 8A 48 98 48 \$F8F2
0B08: A9 00 85 26 A9 54 85 27 \$190C
0B10: A0 00 B1 26 AA BD 00 0C \$8D0C
0B18: 20 ED FD C8 C0 10 D0 F2 \$B2E4
0B20: A4 27 C8 84 27 C0 58 D0 \$FA0A
0B28: E7 20 8E FD A9 54 85 27 \$745A
0B30: A5 26 18 69 10 85 26 C9 \$DCB6
0B38: 00 D0 D5 68 A8 68 AA E8 \$9101

0B40: E8 E8 E8 C8 C8 C8 E0 \$8955
0B48: 60 90 B0 68 A8 88 88 88 \$6121
0B50: 88 C0 9B B0 03 4C BF 9D \$6B44
0B58: A9 8C 20 ED FD 4C AD 0A \$FE76
0B60: 20 7E 0A 8D 84 D0 D2 A3 \$065E
0B68: 00 AD 7A 0A 20 ED FD 20 \$F1F0
0B70: 7E 0A 8D 8D 00 4C AB 0A \$CA9A
BSAVE ULTIMAPPER V, A\$A00, L\$178

MAPPER.OBJ

0C00: 3F 20 20 20 2C 2E 3A 3B \$5BA1
0C08: 3A 54 54 5E 58 2A 2E 2E \$0394
0C10: 41 41 41 41 41 41 41 41 \$1B3C
0C18: 41 41 41 41 3B 3F 3B 3B \$D178
0C20: 2E 2E 2E 2E 2E 2E 2E 2E \$3868
0C28: 3F 3F 3F 3F 3F 3F 3F 3B \$3CDA
0C30: 2E 2E 2E 2E 2E 2E 2E 2E \$54E2
0C38: 3F 41 2B 2B 2B 2B 41 2B \$6FAD
0C40: 3F 3F 3F 3F 3F 3F 3F 3D \$5D8C
0C48: 3F 3F 3F 3F 3F 3F 3F 3F \$4DFC

0C50: 3F 3F 3F 3F 3F 3F 3F 3F \$5D8C
0C58: 3F 3F 3F 3F 3F 3F 47 47 \$F9B8
0C60: 20 20 20 20 20 20 20 20 \$A938
0C68: 20 20 3D 3D 20 20 20 20 \$7CAF
0C70: 3F 3F 3F 3F 3F 3F 3F 3F \$BCAF
0C78: 3F 3F 3F 3F 3F 3F 3F 3F \$7CAF
0C80: 3F 3F 3F 3F 3F 3F 3F 3F \$BCAF
0C88: 3F 3F 3F 3F 3F 3F 3F 40 \$0390
0C90: 3F 3F 3F 3F 3F 3F 3F 3F \$7320
0C98: 3F 3F 3F 3F 3F 3F 3F 3F \$0390

0CA0: 3F 3F 3F 3F 2B 3F 3F 3F \$F170
0CA8: 3F 3F 3F 3F 3F 3F 3F 3F \$21C0
0CB0: 3F 3F 3F 3F 3F 3F 3F 3F \$F170
0CB8: 3F 3F 3F 3F 3F 3F 3F 3F \$21C0
0CC0: 3F 3F 3F 3F 3F 3F 3F 3F \$F170
0CC8: 3F 3F 3F 3F 3F 3F 3F 3F \$21C0
0CD0: 3F 3F 3F 3F 20 20 20 20 \$692C
0CD8: 3F 3F 3F 3F 3F 3B 3F 41 \$5769
0CE0: 3F 3F 3F 3F 3F 3F 3F 3F \$E709
0CE8: 3F 3F 3F 3F 3F 3F 3F 3F \$5769

0CF0: 3F 3F 3F 3F 3F 3F 3F 3F \$E709
0CF8: 3F 3F 3F 3F 3F 3F 3F 3F \$5769
BSAVE MAPPER.OBJ, A\$C00, L\$100

Enhancements

Two drive patch - reads from drive 2 instead of 1.

A5E:B2

AC7:02

Formfeed other than ASCII 12 (Hex \$0C).
B59:XX

Send codes to printer (for neater maps). The last 5 bytes of Ultimapper are 8D 00 4C AB 0A. You can put up to 100 codes (should be adequate) here to support your interface card and printer. Be sure to end your codes with 8D 00 4C AB 0A. The 8D flushes the buffer and activates your codes. The 00 signals that the end of codes has been reached. The 4C AB 0A is a jump back into the program.

I have a Grappler+ and a NX-10 printer. I use the characters from 128-254 plus the normal character set to give me more shapes to print. Unfortunately, the grappler automatically strips off the high-bit. The sequence CHR\$(9)"X" enables the high-bit. Also, when you print a 10X10 square of characters on a printer, it ends up taller than it is wide. To remedy this problem, I use the sequence CHR\$(27)"1" which prints in

James J. Harvey MI

Softkey for...

Perplexing Puzzles

Hartley Courseware, Inc.

This is a game program in deductive reasoning that will help students think more logically, read more critically and question what they read. There is an optional tutorial disk. The program keeps track of each player's progress. All of the puzzles are modifiable.

- 1. Boot DOS 3.3 System Master.
2. Initialize a disk and set it disk aside.
INIT HELLO
DELETE HELLO
3. Put the DOS 3.3 System Master back in the drive.
CALL-151
B942:18
B925:18 60
3D0G
RUN COPYA
4. Copy the original program onto a blank disk.
5. When the copy is finished, put the System Master back in the drive.
BRUN FID
6. Copy all of the files from your copy onto your initialized disk.

a 1X:1Y ratio (7/72 inch linefeed) giving a truly square map. With these commands, I must modify the program this way:
B73:1B 31 09 58 8D 00 4C AB 0A

Now, because I have lengthened the program, I must save to keep the extra bytes.
BSAVE ULTIMAPPER V, A\$A00, L\$17C

After you have the program and index to your liking, then just run the program.
BLOAD MAPPER.OBJ *or your own index*
BRUN ULTIMAPPER V

The program will prompt for printer slot (0 to end) and then ask for you to insert the map disk. (ESC to end). Because the ML program prints so fast, the speed of mapping depends on how fast your printer can print. Afterward, rip the 16 sheets into 4 even sections. Scotch tape the 4 sections together. If you want to highlight certain areas, tape the back first, highlight, then tape the front. I, for instance, highlighted the peaks surrounding each dungeon in the Underworld. This shows how far you can travel from the exit of each dungeon.

To speed printing the map up, I created a index table which would convert the map byte into a printable character. MAPPER.OBJ is that index, but you might want to customize the index to your own liking. The following steps detail the process of creating an index.

1. Enter the monitor:
CALL -151
2. Make all bytes from \$C00 to \$CFF a "?" (an unidentified object).
C00:3F
C01-C00.CFEM

3. Type in the desired character at each location. The following table gives my index character (HEX value first, ASCII character second), the name of the map item, and the location(s) which need to contain the character. For example: if you wanted the docks and bridges to be '-' instead of '=', then you would figure out the hexadecimal value of '-'. The value is 45 which in hexadecimal is \$2D. The table tells you locations 47, 6A, and 6B need to contain \$2D.

C47:2D
C6A:2D
C6B:2D

Map Table

20	Water, Rivers, Falls	1-3, 60-69, 6C-6F, D4-D7
2E	Grass, Trails, Paths	5, E-F, 20-26, 30-37
2C	Swamp	4
3A	Brush	6, 8
54	Forest, Trees	9-A
5E	Foothills	B
58	Mountains	C
2A	Peaks	D
3B	Desert	7, 1C, 1E-1F, 2F, DD
41	Building	10-1B, 39, 3E, DF
2B	Castle Walls, Signs	3A-3D, 3F, A4
3D	Bridges, Docks	47, 6A-6B
47	Guardians	5E-5F
40	Lava	8F
3F	Unknown Items	

4. Repeat this until you have all of the values entered.
5. Save this file.
BSAVE NEW INDEX.OBJ *(or whatever)*, A\$C00, L\$100
6. BLOAD this file before running the mapper.

Oh, by the way. It wasn't too difficult to convert the BASIC program to Machine Language. If anyone out there wants a notated listing of my ML program, send me \$3 with a SASE. And if anyone wants to have me convert a BASIC program of theirs, just write and tell me BASICally what the program is about and how long it is (all the typical stuff.) I am considering working for \$5 per hour (Convenient seeing as I keep track of my own hours.) Don't worry, I wouldn't jip ya, would I?

Mike Maginnis CA

Here's some helpful stuff I dug up rather recently from the huge pile of printouts that sit in the middle of my floor.

Bogus 18 sector disks?

When I received COMPUTIST #67, I of course scanned it, concentrating on the articles that appealed to me the most. The first eye-catcher (aside from the extra-tracks letter which I haven't examined yet) was the bit on uncopiable 18-sector disks by The Saltine. Here's a little bit more info on a protection method that looks like the 18-sector tracks, but which is not (this technique is actually kind of old): the protection being used in the newer Broderbund releases is a very efficient way of storing data on a disk. Each track consists of 6 sectors, and each sector contains 3 pages of memory. (Note, again, that this has been confused by many as an 18-sector track, when it is not.) The reason 18 pages of memory can be stored on track (compared to the 16 pages per track of DOS 3.3) is that there are only 6 sectors per track, which confuses most nibble copiers and fast copiers as well. Between each sector on a track there are sync gaps of FF's, address and data markers and other overhead. By having only 6 sectors, much of this overhead is eliminated. Also the data and address markers are more compacted, and most of the sync bytes

have been removed, so that 18 pages of memory can be crammed on a single track. I'm not sure if The Saltine's methods will work on these disks.

Softkey for...

Eliminator

Adventure International

I can't believe this one's on the most wanted list. Hopefully it's because no one bothered to crack it, not that we're unable.

1. Initialize a DOS 3.3 disk, then:
BLOAD DEMUFFIN PLUS, A\$6000
2. Boot Eliminator and reset into the monitor.
3. Put the newly init'ed disk into drive 2.
803-6000.8000M N 803G *start DeMuffin*
4. Copy all files with the wildcard (=).
5. Boot the Copy.
BLOAD ELIM1
BLOAD STATION
CALL -151
A964:FF
1927:EA EA EA
BSAVE ELIMINATOR, A\$7FD, L\$8300

That's it.

Bitkey for...

Borg

Sirius

Use EDD III to make a bit copy using these parms.
 T1.5-TB.5
 TD-TE
 T0 PARM 28=2 OR 3

DOS 3.3P (for protected)

You'll notice, every now and then, on a disk label, a copyright notice for DOS 3.3P from Apple Computer. This is Apple's protection scheme, the P standing for Protected apparently, and it is only available to the publishers. You will find a lot of software is protected with DOS 3.3P, most of it educational stuff such as The Sesame Street stuff, Special Delivery programs, Spelling Demons, and much more. It seems that Electronic Arts has taken to using a form of it on some of their stuff, though it's obviously protected in other ways as this method can't read from the disk. Perhaps given the proper checksums (or lack thereof) and the proper headers and other RWTS data, this method will work. You'll notice on the label of your Chuck Yeager's AFT this DOS 3.3P copyright notice reading: 'Apple DOS - DOS 3.3/3.P Copyright 1980-1981...', etc. and it is probable that Wasteland uses this protection as well. Simply look for this copyright notice on the disk, packaging or documentation to see if this is being used.

Assuming that this is the only protection used, this program will work for DOS 3.3P. However, some other RWTS and memory modifications might be necessary for this to work. Case in point: Chuck Yeager's AFT. If someone could figure out how to make the disk at least readable, this might work. Here it is:
CALL -151
B6B3:A0 0A B9 D6 B6
B6B8:99 F6 B8 88 10 F7 60 BD
B6C0:8C C0 10 FB 49 AD F0 09
B6C8:BC 8C C0 10 FB B9 00 BA
B6D0:2C A9 00 A0 56 60 20 BF
B6D8:B6 EA EA EA EA EA EA EA
B6E0:EA
BSAVE CRACK DOS 3.3P, A\$B6B3, L\$2E

To use it, just BRUN the program and FID all the files off of the protected disk to a normal one.

Half & quarter Tracks

Another common copy protection scheme is half and quarter tracking. The famous pirate "Parity Error" had this to say about half and quarter tracking techniques:

"Apple disks have data written in 35 concentric circles known as tracks. This is common knowledge. But these tracks are written approximately 1/48th of an inch apart. It is possible to write closer than this, but it will most likely overwrite some of the data on the adjacent track(s). However, the Apple disk drive's head can be positioned over 1/96th of an inch increments. Because standard copiers copy only 'full-tracks', and not 'half-tracks', it is sometimes used as a protection scheme. Most decent nibble/bit copiers such as Locksmith 6.0 and Copy II+ or EDD have the ability to allow you to copy these half tracks, but what if you want to be able to do this in your own programs? That's what this is for. I'm sure some enterprising cracker out there can figure out how to use these routines in a program like Super IOB.

"Parity Error" continues: (In copying half and quarter tracks) the final problem is that the foreign RWTS will be looking for data on half-tracks, and not finding it there. You need to tell their RWTS that the tracks are now full-tracks. There are several softswitches that need to be accessed for any use of the disk drive:

SWT	Label	What it does
C080	phase off	stepper motor 0 phase off
C081	phase on	stepper motor 0 phase on
C082	phase 1 off	stepper motor 1 phase off
C083	phase 1 on	stepper motor 1 phase on
C084	phase 2 off	stepper motor 2 phase off

C085	phase 2 on	stepper motor 2 phase on
C086	phase 3 off	stepper motor 3 phase off
C087	phase 3 on	stepper motor 3 phase on
C088	motor off	turns off disk drive
C089	motor on	turns on disk drive
C08A	drive 0 en	engages drive 0 (1)
C08B	drive 1 en	engages drive 1 (2)
C08C	q6i-strobe data latch for I/O	reads a byte
C08D	q6h-Read data latch	senses write protect
C08E	q7i-prepare latch for input	writes a byte
C08F	q7h-prepare latch for output	loads write latch

Normal DOS moves the head from track 0 to track 1 in the following manner:

- a) turn on phase 1 (track .5) and wait
- b) turn off phase 0 (track 0) and wait
- c) turn on phase 2 (track 1) and wait
- d) turn off phase 1 (track .5) and wait
- e) turn off phase 2 (track 1) and wait
- f) return to the caller

By examining the above example, you can see what is needed to move the head the half-tracks."

This little patch to DOS will allow you to do the job without all the manual look-ups and calls. This is especially useful if you are going to use RWTS directly to convert a protected disk to a standard one.

The Apple drive stepper motor requires 2 pulses to move one whole track, so if we send out one more pulse, we will move the arm 1/2 a track beyond the normal position for that track, landing on a half-track.

The routine that positions the head is located in DOS starting at \$B9A0 so right there we make it JSR to our half-track routine which we will put at \$300 for our purposes, but which can be put anywhere that isn't overwritten by something else. So:

CALL-151
B9A0:20 00 03 EA
0300:86 2B 85 2A C9 xx 90 02
0308:E6 2A AD 78 04 C9 xx 90
0310:03 EE 78 04 A5 2A 60

Where xx is the hex value equal to twice the track number of the first track you want to half-track. Remember you need two pulses per track. After you have read the track and are ready to write it out to a whole track, just change \$304 from C9 to 60 so that the routine is bypassed and the track will be written out as a whole track. If you want to relocate the routine, fine but replace bytes \$B9A1-\$B9A2 with the address where the routine will be located. Remember that m/l stores addresses in reverse (i.e. address 0300 is translated as 00 30).

Experiment and enjoy, and as always, send in everything you find out to COMPUTIST so the rest of us can share the knowledge.

Bob Igo PA

To The Switch and Michael Warren: I'd like someone to speak in the APT language with. My address was in Computist #63, so if you can, please contact me.

To Tim Valuk: Joe Montano gave us his Bard's Tale III editor for free in Computist #64. Besides, you would be supporting the magazine by sending in your programs instead of trying to make money from them. The staff will decide whether or not your program is too large.

Advanced Playing Technique for...

Chrono Warrior

For extended lives (recommended value of ?? is from 06 to 32, although larger values may work.)

Trk	Sci	Byte(s)	From	To
\$04	\$00	\$08	05	??

Advanced Playing Technique for...

Anti Gravity

For unlimited lives:

Trk	Sci	Byte(s)	From	To
\$20	\$02	\$24-26	CE 09 60	EA EA EA
\$20	\$04	\$D6-D8	CE 09 60	EA EA EA

Advanced Playing Technique for...

Demonic Decks

For unlimited lives:

Trk	Sci	Byte(s)	From	To
\$13	\$03	\$29-2A	C6 F1	EA EA

For extended lives replace ?? with values from 00 to FF.

Trk	Sci	Byte(s)	From	To
\$13	\$07	\$D2	03	??

Softkey for...

Pool of Radiance

SSI

Make this edit to patch the code wheel check on side one so you just hit return.

Trk	Sci	Byte(s)	From	To
\$12	\$0A	\$47-99	?	all 00's

For those of you who have requested an explanation of this technique, I can tell you a little

bit as I only know a little bit. Now, if you've seen the code wheel on this game, you know how annoying it is to play around with it just to get on with the game. I assumed that the words on the code wheel would probably be on the disk in some way. There are six concentric lines of text on the wheel. The inner wheel starts at 1 and goes to 9, then A-Z. So, I thought that maybe they store that in memory along with the other lines and decode it to check it with the wheel. Thankfully, this was not the case. I searched for a code name, WYVERN. This appeared on track \$12, sector \$0A along with all the other code names. "Hmmm...", I thought out loud, "Maybe I can find the part of the program that's checking your input with the acceptable code name. How? Maybe I can use Crucial Code Finder from the ancient depths of Hardcore Computing to scan memory for the name."

I can't remember if I found the name or not, but if I did, I couldn't find the input routine. So, I turned all the names into 00 with my sector editor. Since each word was separated with 00, putting 00 everywhere wouldn't load any data into memory. Now, just hit return, giving it nothing which equals what is in its memory when it asks you for the name. Sorry I couldn't be more specific, but this is my first code wheel elimination. Maybe if companies do go into this form of copy protection, I'll finally be able to crack more disks.

Softkey for...

Times of Lore

Origin Systems

To give yourself a magical axe

Trk	Sci	Byte(s)	From	To
\$22	\$00	\$48	\$10	\$15

Advanced Playing Technique for...

Pool of Radiance

SSI

Here's some of the edit information I've found on Pool of Radiance. I located where my characters were being stored by using Locksmith 6.0's 16-sector compare utility. That told me the differences between saved games each done a few minutes after the other. I'm not sure if there's a specific place where the characters are stored, but be sure to modify the saved ones and not the ones on the roster.

To find where they are, save the game once you have some experience and (S)can for (T)ext on the Copy II+ sector editor. Scan for the lead character's name. Check (A)ddresses E8 and E9. If they're not zeroes, this is your saved character. If they are zeroes, the character is the one on the roster. On my game, saved characters began on track \$02, sector \$0E. Of course, you can modify the ones on the roster, but it's usually more practical to change the saved ones. Anyway, here's what I've found so far.

Character Information:

At the top of the sector (bytes \$00-\$13, I think) is where your character's name is stored. Text values are \$80 less than normal, so a "G" would be \$47, not \$C7. (The hi-bit is cleared.)

byte	What it is
14	strength
15	intelligence
16	wisdom
17	dexterity
18	constitution
19	charisma
All of the above may range up to \$15 safely. Try higher values experimentally if you so desire.	
1A	exceptional strength percentage: This ranges from \$00-\$64. I haven't tried higher values as the actual AD&D game only goes up to 100% with regards to exceptional strength.
76	maximum hit points: Go for about \$E0 because resting repeatedly will reset the current hit points to \$00 which is quite annoying.
9A-9E	saving throws: Low values here are desirable.
A0	character level
BA-BC	copper
BD-BE	silver
BF-C0	electrum
C1-C2	gold
C3-C4	platinum
C5-C6	gems
C7-C8	jewelry

I'm not real sure about the money. E8-E9 experience points

Values reset themselves to minimum for the current level after training with much more than enough experience to advance. It's not like BT III where you could raise as many levels as you could.

EE-F6 spell memorizations for magic users, 1st-9th level respectively. Try values around \$0F. That should suffice. Although you can edit the spell memorizations for all the levels, only spells of level 1-3 are used in this particular module of the game.

Item information

I've found item codes for about 252 items, but I'm not sure how they're encoded. I know enough that I can turn mundane objects into magical ones

and give charged items more charges. So, if you like your wand a whole lot, just zap it with my edits and you'll always have it with periodic checkups.

Generally, this is what I know about the item code bytes:

- Byte What it is**
- 00 1st object phrase (such as Two-handed Sword)
 - 01 3rd object phrase (such as +/3 vs. Undead)
 - 02 2nd object phrase (such as +1)
 - 03 1st object phrase again (sometimes)
 - So, above, we have a character with a Two-handed Sword +1/+3 vs. Undead.
 - 04 Actual plus of weapon/armor up to \$7F, or 127. Higher numbers somehow turn negative. (i.e. \$80 = -128)
 - 05 Charges of the item if it has them.
 - 06 80=equipped item, 00=unequipped
 - 07 no clue
 - 08 no clue
 - 09 no clue
 - 0A quantity of the item
 - 0B no clue
 - 0C no clue
 - 0D no clue
 - 0E no clue
 - 0F no clue

After I list the object codes for you, I'll show you how to turn an ordinary weapon into a devastating one and how to lower your armor class by using very magic armor. To find out exactly where the possessions are stored, use the above table and search for a known possession. My characters' possessions were on track \$04, sectors \$02-\$08.

- | | |
|---------------------------|--------------------------------|
| 01 = battleaxe | 02 = hand axe |
| 03 = bardiche | 04 = bec de corbin |
| 05 = bill-guisarme | 06 = bo stick |
| 07 = club | 08 = dagger |
| 09 = dart | 0A = fauchard |
| 0B = fauchard-fork | 0C = flail |
| 0D = military fork | 0E = glaive |
| 0F = glaive-guisarme | 10 = guisarme |
| 11 = guisarme-voulge | 12 = halberd |
| 13 = lucern hammer | 14 = hammer |
| 15 = javelin | 16 = jo stick |
| 17 = mace | 18 = morning star |
| 19 = partisan | 1A = military pick |
| 1B = awl pike | 1C = quarrel(s) |
| 1D = ransour | 1E = scimitar |
| 1F = spear | 20 = spetum |
| 21 = quarter staff | 22 = bastard sword |
| 23 = broad sword | 24 = long sword |
| 25 = short sword | 26 = 2-handed sword |
| 27 = trident | 28 = voulge |
| 29 = composite long bow | 2A = composite short bow |
| 2B = long bow | 2C = short bow |
| 2D = heavy crossbow | 2E = light crossbow 2F = sling |
| 30 = mail | 31 = armor |
| 32 = leather | 33 = padded |
| 34 = studded | 35 = ring |
| 36 = scale | 37 = chain |
| 38 = splint | 39 = banded |
| 3A = plate | 3B = shield |
| 3C = wood | 3D = arrow(s) |
| 3E = ??? | 3F = ??? |
| 40 = potion | 41 = scroll |
| 42 = ring | 43 = rod |
| 44 = stave | 45 = wand |
| 46 = jug | 47 = amulet |
| 48 = apparatus | 49 = bag |
| 4A = beaker | 4B = boat |
| 4C = book | 4D = boots |
| 4E = bowl | 4F = bracers |
| 50 = brazier | 51 = brooch |
| 52 = broom | 53 = purse |
| 54 = candle | 55 = carpet |
| 56 = censer | 57 = chime |
| 58 = cloak | 59 = crystal |
| 5A = cube | 5B = cubic |
| 5C = fortress | 5D = decanter |
| 5E = deck | 5F = drums |
| 60 = dust | 61 = eyes |
| 62 = figurine | 63 = flask |
| 64 = gauntlets | 65 = gem |
| 66 = girdle | 67 = helm |
| 68 = horn | 69 = horseshoes |
| 6A = incense | 6B = stone |
| 6C = instrument | 6D = javelin |
| 6E = jewel | 6F = ointment |
| 70 = libram | 71 = lyre |
| 72 = manual | 73 = mattock |
| 74 = maul | 75 = medallion |
| 76 = mirror | 77 = necklace |
| 78 = net | 79 = pigment |
| 7A = pearl | 7B = periapt |
| 7C = phylactery | 7D = pipes |
| 7E = hole | 7F = token |
| 80 = robe | 81 = rope |
| 82 = rug | 83 = saw |
| 84 = scarab | 85 = spade |
| 86 = sphere | 87 = stone |
| 88 = talisman | 89 = tome |
| 8A = trident | 8B = grimoire |
| 8C = well | 8D = wings |
| 8E = vial | 8F = lantern |
| 90 = mirror | 91 = flask of oil |
| 92 = 10' pole | 93 = 50' rope |
| 94 = iron | 95 = thieves' picks & tools |
| 96 = iron rations | 97 = standard rations |
| 98 = holy symbol | 99 = vial of holy water |
| 9A = vial of unholy water | 9B = barding |
| 9C = dragon | 9D = lightning |
| 9E = saddle | 9F = small raft |
| A0 = cart | A1 = wagon |
| A2 = +1 | A3 = +2 |
| A4 = +3 | A5 = +4 |
| A6 = +5 | A7 = of |
| A8 = ??? | A9 = cloak |
| AA = displacement | AB = torch(es) |
| AC = oil | AD = speed |
| AE = tapestry | AF = bodily health |
| B0 = copper | B1 = silver |
| B2 = electrum | B3 = gold |

- | | |
|---------------------------|--------------------------|
| B4 = platinum | B5 = ointment |
| B6 = keogthum's | B7 = sheet(s) |
| B8 = strength | B9 = healing |
| BA = holding | BB = extra |
| BC = gaseous form | BD = slipperiness |
| BE = jewelled | BF = flying |
| C0 = treasure finding | C1 = fear |
| C2 = disappearance | C3 = statuette |
| C4 = fungus | C5 = chain(s) |
| C6 = pendant | C7 = broach |
| C8 = of seeking | C9 = -1 |
| CA = -2 | CB = -3 |
| CC = lightning bolt | CD = fire resistance |
| CE = magic missiles | CF = save |
| D0 = clerical scroll | D1 = MU scroll |
| D2 = with 1 spell | D3 = with 2 spells |
| D4 = with 3 spells | D5 = protection scroll |
| D6 = jewelry | D7 = fine |
| D8 = huge | D9 = bone |
| DA = brass | DB = key |
| DC = AC2 | DD = AC6 |
| DE = AC4 | DF = AC3 |
| E0 = of protection | E1 = paralyzation |
| E2 = Ogre power | E3 = invisibility |
| E4 = missiles | E5 = elvenkind |
| E6 = rotting | E7 = covered |
| E8 = efreeti | E9 = bottle |
| EA = missile attractor | EB = of maglubiyet |
| EC = secr door & trap det | ED = good dragon control |
| EE = leather falling | EF = giant strength |
| FO = restoring level(s) | F1 = flametongue |
| F2 = fireballs | F3 = spiritual |
| F4 = boulder | F5 = diamond |
| F6 = emerald | F7 = opal |
| F8 = sapphire | F9 = of Tyr |
| FA = of Woodens | FB = of Sune |
| FC = wooden | FD = +3 vs. undead |
| FE = pass | FF = ??? |

So, the Two-handed Sword +1/+3 vs. Undead would have approximately the following code:
26 FD A2 26 7F 00 80 00
00 00 01 00 00 00 00 00

Actually, here we have an equipped, Two-handed Sword +1/+3 vs. Undead which is actually a +127 weapon. To find out what the unknown values are, scan disk #3 for the item codes. You've found them when you see something like this:
01 XX XX ...
XX XX XX ...
02 XX XX ...
XX XX XX ...
03 XX XX ...
etc.

Then, write them down for something like a shield, 3B. Record that information on the sector where your character's possessions are. Change relative byte 04 to anything up to and including 7F and you'll be pretty well protected. You can do the same for armor (use leather armor if you must as it weighs less.) You can also turn something as lowly as a dagger into a demon skewer with a similar edit to relative byte 04.

I'm not sure I'll do more work on this information because it's more than enough to help with the game. Although THAC0 and current hit points still elude me, I'm happy to have some characters with armor classes of -120 or more.

Please note that in all cases of adventure games, I play the game straight with normal characters, only using edits on rare occasions, such as the last battle in Bard's Tale III or to defeat the Trolls on POR. However, to test my APT information, I run a separate game with my "illego"s. Admittedly, it can be satisfying for a while when nothing can touch you, but after a short time, it becomes overkill. I like to know how things work, and my APT's are the product of my research. I just wanted the readers of APT's to know that I'm not incompetent when it comes to games. The only person you're cheating is yourself when you never play games without the edits.

P.S. If anyone would like an explanation of how I made a POR quick disk to take the place of all disks when you have to insert one side of each one, let me know by writing or by sending an article to the magazine. If there is enough demand, I shall write such an article.

John Jackson CA

Softkey for...
Who, What, Where, When, Why
Language Arts/Hartley

1. Use COPYA without error checking.
LOAD COPYA
POKE 47426,24
RUN

That's all. It's now COPYA-able.

Softkey for...
Gnee or Not Gnee
Sunburst

1. Boot original and break into the Monitor.
1900<B800.BFFF
Move RWTS
2. Boot a slave disk
BSAVE GNEE.RWTS, A\$1900, L\$800 on disk with SuperJOB
LOAD SUPER.JOB
EXEC SWAP.CON
3. Fix line 10010 to load GNEE.RWTS.
RUN

Mine made noisy errors on track 1E, but copy continued OK.

Softkey for...
Moptown Hotel
Learning Co.

1. Boot original and break into the Monitor.
1900<B800.BFFF
Move RWTS
2. Boot a slave disk.
BSAVE MOP.RWTS, A\$1900, L\$800 on disk with SuperJOB
LOAD SUPER.JOB
EXEC SWAP.CON
3. Fix line 10010 to load MOP.RWTS.
RUN

Softkey for...
Mystery Objects
M.E.C.C.

1. Use COPYA without error checking.
LOAD COPYA
POKE 47427,28
RUN
2. Use Copy II Plus to copy a clean ProDOS to the copy.
3. Search for 90 03 4C ?? ?? 60 and change to 18 EA EA EA EA 60. (Track \$01, sector \$09, byte \$26, on my disk.)
4. Write changes back to disk.

Softkey for...
Mystery Matter
M.E.C.C.

1. Use COPYA without error checking.
LOAD COPYA
POKE 47427,28
RUN
2. Use Copy II Plus to copy a clean ProDOS to the copy.
3. Search disk for 90 03 4C ?? ?? 60 and change to 18 EA EA EA EA 60. (Track \$05, sector \$09, byte \$68, on my disk.)
4. Write changes back to disk.

Softkey for...
Wood.Car.Rally
M.E.C.C.

1. Use COPYA without error checking.
LOAD COPYA
POKE 47427,28
RUN
2. Use Copy II Plus to copy a clean ProDOS to the copy.
3. Search disk for 90 03 4C 2F 85 60 and change to 18 EA EA EA EA 60. (Track \$01, sector \$09, byte \$04, on my copy.)
4. Write changes back to disk.

Troi Miller NC

I have an Apple II GS with approximately 2 meg of memory. I am very interested in learning how to move most of my 3.5 software onto my Applied Ingenuity hard drive. I read the article by Tim Valuk on how he unprotected 4th and Inches. I tried to apply his technique to Accolade's The Duel, IIGS version. I found the "A2 20 A0 01" but the preceding bytes were "E2 30" not "C2 30" as in his case. I tried replacing the "A2 20" with "18 6B". The copy booted successfully as a 3.5 but crashed with a ProDOS error 201 when I moved the program onto the hard disk.

I successfully managed to make a copy of Marble Madness, IIGS. However I still cannot move it onto the hard disk. I searched for various text strings and found lots of references to the folder name on the disk. I tried changing the references to a folder on my hard disk. I destroyed the copy. Oh well. Anybody have any suggestions?

Gustavo Delfino FL

Installing GS/OS on Sierra 3D Adventures

Requirements:
Apple IIGS
Rom 01
Deprotected Sierra 3D Game.
System Disk 4.0 or 5.0

Sierra games comes with the old and slow ProDOS 16, and doesn't have enough space to install GS/OS. But Sierra games doesn't use all tools, you can delete the following tools from System/Tools Subdirectory: Tool 18, Tool 19, Tool 22, Tool 27. Also you can delete the Fonts and Desk.Acce subdirectories.

1. Delete Tools 18, 19, 22, 27; from your Sierra 3D Game startup disk.
2. Delete the Fonts and Desk.Acce subdirectories.

3. Delete the ProDOS and P16 files.
4. Create A subdirectory inside the system called FSTS
5. Copy from your System disk 4.0 the following files to the same allocation in your sierra game disk:
/System.Disk 4.0/
ProDOS
/System.Disk 4.0/System/
Start.GS.OS
GS.OS
Error.MSG
/System.Disk 4.0/System/Drivers/
Apple Disk 3.5
/System.Disk 4.0/System/FSTS/
Char.FST
Pro.FST
/System.Disk 4.0/System/Tools/
Tool 14
Tool 15
Tool 16
Tool 20
Tool 21
Tool 23
Tool 25
Tool 28

6. Format a Disk in 800K 2:1 using Finder 1.2
7. File copy your sierra startup game to the "2:1" formatted disk.

If you have a GS-RAM expansion card, I recommend you to install AE.CACHE.SETUP instead of GS/OS. The boot time will be longer than using GS/OS (v4.0), but the game works faster.

A note about the magazine

If there is no money to print the magazine, don't print it, sent it on 5.25 disks, or for extra money on a 3.5 disk on an Standard ProDOS text file. And then we can print the magazine in our home, with our paper, our ink, and our electricity.

A nice idea but with a slight flaw. It costs more to produce the newsletter on disk than on paper. Take a look at the high prices for other disk-azine type publications.RDEXed

Mr. E WI

In the two softkeys listed below I have tried to do more than just tell you how to copy a particular disk. Instead I have tried to set up a "formula" to follow for deprotecting disks that are copy protected in the same fashion as these two programs are. The first is protected with a bad block, and the second is protected with the 20/21 nibble count. Since these are two of the most popular protection schemes used on GS programs, hopefully, this will allow you to not only deprotect these disks, but also allow you to deprotect many others in addition. Good Luck!

Softkey for...
Calendar Crafter v1.2
MECC

Requirements:
Apple IIGS 512K
3.5" disk copier
3.5" sector editor (I used Copy II Plus v8.4)

MECC is up to version 1.2 of Calendar Crafter, and with the new version MECC has moved the location of its copy protection. The protection, however, remains the same as in the previous releases: a check for a bad block.

I personally find that a disk protected with a bad block is probably the easiest form of protection to remove. After making a copy of the protected disk (you will get a read error), the next step to take is to install the Classic Desk Accessory Visit Monitor and boot the copy.

- 1) Enter the monitor (CALL -151).
- 2) At the * prompt type # and press return.
- 3) Insert the copy and press ctrl openapple reset.

When the program asks for the original disk, enter the control panel by pressing ctrl openapple esc and select Visit Monitor. Now use the monitor's search command to locate the sequence 22 A8 00 E1 22 (the ProDOS 16/GSOS read block command).

- 1) At the * prompt type /22 A8 00 E1 22/<n/0.FFFFF, where n is the bank number to be searched, and press return.
- 2) Begin with n=1 (22 A8 00 E1 22/<1/0.FFFFF) and keep incrementing n by one until the search is successful.
- 3) If the screen changes, you can get back to the normal text screen by pressing ctrl T followed by a return. You can shut off 5.25" disk drives by typing COE8 and pressing return.

Once the sequence is located, the m and x flags must be set to zero so that you can list the code in proper disassembly format using the L command.

- 1) At the * prompt type 0=m and press return.
- 2) At the * prompt type 0=x and press return.

Now begin listing the code at the address or addresses turned up in the search.

Following this procedure on the Calendar Crafter disk showed that 22 A8 00 E1 22 was located four times in bank \$01, beginning at address \$A8A2. Listing this code revealed the following:

A8A2:22 A8 00 E1JSL E100A8 Read block before bad block
 A8A6:22 00 0022
 A8A8:1F A9 01 00 0001A91F
 A8AC:90 0F BCC A8BD (+0F) Clear carry = no bad block
 A8AE:9C 25 A9 STZ A925
 A8B1:22 A8 00 E1JSL E100A8 Read bad block
 A8B5:22 00 0022
 A8B7:1F A9 01 00 0001A91F
 A8BB:90 02 BCC A8BF (+02) Clear carry = no bad block
 A8BD:18 CLC Everything's OK, clear carry,
 A8BE:60 RTS and return to sender
 A8BF:38 SEC Something's wrong, set carry,
 A8C0:60 RTS and return to sender

01/A8C1: through 01/A8EA: contains some trivial code. Then, picking up at 01/A8EC:
 A8EC:22 A8 00 E1 JSL E100A8 Read block before bad block
 A8F0:22 00 0022
 A8F2:1F A9 01 00 0001A91F
 A8F6:B0 26 BCS A91E (+26) Carry set = bad block
 A8F8:A2 13 00 LDX #0013
 A8FB:BD 29 AB LDA AB29,X Compare what was read
 A8FE:DD 29 A9 CMP A929,X
 A901:D0 1A BNE A91D Not equal = something wrong

A903:CA DEX
 A904:CA DEX
 A905:10 F4 BPL A8FB (-0C)
 A907:CE 25 A9 DEC A925
 A90A:22 A8 00 E1JSL E100A8 Read bad block
 A90E:22 00 0022
 A910:1F A9 01 00 0001A91F
 A914:90 07 BCC A91D (+07) Carry clear = no bad block
 A916:C9 27 00 CMP #0027 Check for I/O error
 A919:D0 02 BNE A91D (+02) Not equal = no I/O error
 A91B:18 CLC Everything's OK, clear carry,
 A91C:60 RTS and return to sender
 A91D:38 SEC Something's wrong, set carry,
 A91E:60 RTS and return to sender

By now you can probably guess how to overcome the bad block check. From the above code we see that Calendar Crafter basically answers only "yes or no" as to whether or not the disk in the drive is the original. If the first block Calendar Crafter checks is legitimate and the second is a bad block (an I/O error occurs), Calendar Crafter answers "yes" by clearing the carry before returning. If either of these two conditions are not present, Calendar Crafter answers "no" by setting the carry before returning. Therefore, the simplest way to remove the copy protection is to replace both the SECs with CLCs. Then, no matter what happens, Calendar Crafter will always clear the carry before returning, thus signifying that the original disk is in the drive.

Once you have figured out how to overcome the protection all that remains to be done is to make the changes to the copy.

- 1) Boot up your favorite sector editor.
- 2) Scan the disk for 22 A8 00 E1 22.
- 3) Make any necessary changes and write the block back to the disk.

On the Calendar Crafter disk the sequence was found in block \$4A9, and the bytes that need to be changed were \$136 and \$194 both from 38 to 18.

Cookbook Method

1. Make a copy of the disk (ignoring any read errors).
2. Make the following changes to the copy:

Blk	Byte(s)	From	To
\$4A9	\$136	38	18
	\$194	38	18

3. Write the block back to the copy.

Softkey for...

Where in the World is Carmen Sandiego

Broderbund

Requirements:

Apple IIGS 512K
 3.5" disk copier
 3.5" sector editor (I used Copy II Plus v8.4)

Broderbund has created a IIGS specific version (with improved graphics and sound) of their very popular program Where In The World Is Carmen SanDiego. The disk, unfortunately, is copy protected with the ever popular 20/21 nibble count, but not for long!

First make a copy of both disks. Next, install the Classic Desk Accessory Visit Monitor and boot the copy.

- 1) Enter the monitor (CALL -151).
- 2) At the * prompt type # and press return.
- 3) Insert the copy and press ctrl openapple reset.

Eventually the program will ask you to insert the master disk. (This occurs when the game disk is in the drive; therefore, we know the protection lies on the game disk.) At this point you want to enter the control panel by pressing ctrl openapple esc and select Visit Monitor. Now you can use the monitor's search command to locate the nibble count routine. Search for either A2 20

A0 01 or A2 21 A0 01.

- 1) At the * prompt type /A2 20 A0 01/<n/0.FFFFP, where n is the bank number to be searched, and press return.
- 2) Begin with n=1 (/A2 20 A0 01/<1/0.FFFFP) and keep incrementing n by one until the search is successful.
- 3) If the screen changes, you can get back to the normal text screen by pressing ctrl T followed by a return. You can shut off 5.25" disk disk drives by typing COE8 and pressing return.

Once the sequence is found, list the code at the address found. Work your way back to the beginning of the routine (until you find a RTS(60) or RTL(6B)).

Following this procedure with the Carmen SanDiego disk came up with the following in bank \$01 beginning at address \$2231:

2231:08 PHP
 2232:E2 30 SEP #30 Set 8-bit accumulator
 2234:A2 20 LDX #20 Track \$20
 2236:A0 01 LDY #01 Side one
 2238:8B PHB
 2239:4B PHK
 223A:AB PLB
 223B:20 7B 22 JSR 227B Go do the nibble count
 223E:D0 23 BNE 2263 (+23) Not equal = read error
 2240:8A TXA
 2241:8D B4 22 STA 22B4 Store first half of result
 2244:98 TYA
 2245:8D B5 22 STA 22B5 Store second half of result
 2248:0D B4 22 ORA 22B4
 224B:F0 16 BEQ 2263 (+16) Results shouldn't be equal

01/224D:through 01/2263:basically repeats the above code except it is now reading track \$21 instead of \$20. Then, picking up at 01/2263:...

2263:48 PHA Come here if something's wrong
 2264:A9 33 LDA #33
 2266:8D B4 22 STA 22B4
 2269:8D B5 22 STA 22B5 Mess up the nibble count results by setting them all equal to \$33
 226C:8D B7 22 STA 22B7
 226F:8D B8 22 STA 22B8
 2272:68 PLA
 2273:82 02 00 BRL 2278 (+02) Branch to return to sender
 2276:A9 00 LDA #00 Load a pass value
 2278:AB PLB
 2279:28 PLP
 227A:6B RTL return to sender

There are two important pieces of information you need to get from this listing. They are the address the routine begins at and how the routine is exited. In this example the routine begins at 01/2231 and is exited by a RTL. A RTL signifies that the routine is called by a JSL, while a RTS would mean the routine is called by a JSR. The next step, then, would be to again use the monitor's search command to find this call (the source of the protection).

- 1) At the * prompt type /22 low byte high byte bank number/<n/0.FFFFP for a JSL, or /20 low byte high byte/<n/0.FFFFP for a JSR, where n in each case is the bank number to be searched, and press return.
- 2) Begin with n=1 (/22 31 22 01/<1/0.FFFFP) and keep incrementing n by one until the search is successful.
- 3) If the screen changes, you can get back to the normal text screen by pressing ctrl T followed by a return. You can shut off 5.25" disk disk drives by typing COE8 and pressing return.

Once the sequence is located, the m and x flags must be set to zero so that you can list the code in proper disassembly format using the L command.

- 1) At the * prompt type 0=m and press return.
- 2) At the * prompt type 0=x and press return.

Now begin listing the code at the address or addresses turned up in the search.

On the Carmen SanDiego disk the sequence 22 31 22 01 was located in bank \$03 beginning at address \$4D22, and the code looks as follows:

4D22:22 31 22 01 JSL 012231 Go to the nibble count
 4D26:22 94 22 01 JSL 012294 Get first half of result
 4D2A:85 EA STA EA Store it
 4D2C:22 9C 22 01 JSL 01229C Get second half of result
 4D30:85 E8 STA E8 Store it
 4D32:A5 EA LDA EA Load first half
 4D34:38 SEC
 4D35:E9 40 1F SBC #1F40 Compare to high end
 4D38:70 03 BVS 4D3D (+03)
 4D3A:49 00 80 EOR #8000
 4D3D:10 31 BPL 4D70 (+31) Too high, failed check
 4D3F:A5 EA LDA EA Load first half again
 4D41:38 SEC
 4D42:E9 D0 20 SBC #20D0 Compare to low end
 4D45:F0 07 BEQ 4D4E (+07) Equal = it's O.K.
 4D47:70 03 BVS 4D4C (+03)
 4D49:49 00 80 EOR #8000
 4D4C:30 22 BMI 4D70 (+22) Too low, failed check
 4D4E:A5 E8 LDA E8 Load second half
 4D50:38 SEC
 4D51:E9 B0 1D SBC #1DB0 Compare to high end
 4D54:70 03 BVS 4D59 (+03)
 4D56:49 00 80 EOR #8000
 4D59:10 15 BPL 4D70 (+15) Too high, failed chk

4D5B:A5 E8 LDA E8 Load second half again
 4D5D:38 SEC
 4D5E:E9 78 1E SBC #1E78 Compare to low end
 4D61:D0 03 BNE 4D66 (+03) Not equal = it's too low
 4D63:82 4D 00 BRL 4DB3 (+4D) Checks passed, everything's O.K.
 4D66:70 03 BVS 4D6B (+03)
 4D68:49 00 80 EOR #8000
 4D6B:30 03 BMI 4D70 (+03) Too low, failed check
 4D6D:82 43 00 BRL 4DB3 (+43) Checks passed, everything's O.K.

The fail code begins at 03/4D70. The continue code begins at 03/4DB3.

If you follow through the code you see that the program goes to the nibble count, comes back, loads in the results of the nibble count, and compares the values to high and low end benchmarks. If at any time a value is either too high or too low, the program branches to \$4D70 where the program will ask for the original disk to be inserted. If all the checks pass, however, the program branches long over the fail code and continues on its merry way. Therefore, to remove the protection we will change the JSL to the nibble count routine at \$4D22 to a BRA (branch always) \$4D6D. This will force the program to skip over the nibble count and comparisons right to the BRL to the continue code.

Once you have figured out how to overcome the protection all that remains to be done is to make the changes to the copy.

- 1) Boot up your favorite sector editor.
- 2) Scan the disk for the call to the nibble count routine (22 31 22 01 on the Carmen SanDiego game disk).
- 3) Make any necessary changes and write the block back to the disk.

With the Carmen SanDiego disk, though, something will happen that is usually not the case. 22 31 22 01 will not be found. Broderbund has encoded the address portion, and it appears as 22 12 02 on the disk. This being the case, all that needs to be done is to search for a different sequence from further down in the code (49 00 80 10 for instance) and work your way back. Now that the proper code is located you are ready to make your changes.

On the Carmen SanDiego disk the code was found in block \$67 and the bytes that needed to be changed were \$59-60 from 22 12 to 80 2A. Initially, you'd think you'd want to make it 80 49, but you must remember that the second byte is encoded. \$31 - \$12 = \$1F so \$49 - \$1F = \$2A.

Cookbook Method:

1. Make a copy of both disks.
 2. Make the following changes to the game disk copy:
- | Blk | Byte(s) | From | To |
|------|---------|-------|-------|
| \$67 | \$59-60 | 22 12 | 80 2A |
3. Write the block back to the copy.

Bitkey for...

The Hunt for Red October

DataSoft

Requirements:

Apple IIGS 512K
 Copy II Plus (v8.4)

The Hunt For Red October is another disk along the lines of Alien Mind. The information is stored on disk in a special compressed format. As you probably know, this makes deprotecting the disk extremely difficult. Unlike Alien Mind, however, you can bit copy the disk very easily. Just use Copy II Plus and go to 3.5" bit copy.

Raymond Karl PA

Softkey for...

Platoon

Data East

Requirements:

COPYA!
 Sector Editor (I used Copy II+)

Copy both sides of Platoon with COPYA. On side A only, edit:

Trk	Sec	Byte(s)	From	To
00	0E	4A	4C 00 C6	EA EA EA

Thanks to the various contributors whose articles provided me with enough information to crack Platoon. I would appreciate it if someone would send in a softkey for Border Zone by InfoCom.

Bad Bob CA

IIGS Softkey for...

Xenocide

Micro Revelations, INC.

Requirements:

1 blank 3-1/2" disk
 Prosel's Block Warden

I saw a post by Brian A. Troha that had a softkey for XENOCIDE. It appears there are a few versions of this out, as his softkey did not work on my version, but this is what worked for mine. Use PROSEL's Block Warden and (F)ollow

/XENOCIDE/XENO.SYS16 and make the following changes:

Relative Byte	Original value	Change to	Relative Block
\$155	22 72 09 00	AF 72 09 00	1
\$159	90 0C	80 0A	
\$15B	22 72 09 00	AF 72 09 00	
\$15F	90 06	80 04	
\$161	22 8B 1D 00	AF 8B 1D 00	
\$165	80 EE	EA 18	
\$78FC	00 00 8B	46 1E 8B	61
\$7901	A9 00 00	A9 46 1E	
\$7A25	22 72 09 00	AF 72 09 00	62
\$7A29	90 0C	80 0A	
\$7A2B	22 04 00 00	AF 04 00 00	
\$7A2F	90 06	80 04	
\$7A31	22 8B 1D 00	AF 8B 1D 00	
\$7A35	80 EE	EA 18	
\$8779	F0 01 60	EA EA EA	68
\$B58C	22 04 00 00	AF 04 00 00	91
\$B590	90 0C	80 0A	
\$B592	22 04 00 00	AF 04 00 00	
\$B596	90 06	80 04	
\$B598	22 8B 1D 00	AF 8B 1D 00	
\$B59C	80 EE	EA 18	
\$C5B4	D0 27	EA EA	99
\$F89D	8F	AF	125

And some cheats to make life easier and everlasting!

Unlimited	Rel Byte	From	To
Men	\$ 4DF	CE 1C 00	AD 1C 00
Fuel	\$1774	CE 99 15	AD 99 15
Shields	\$17B0	8D 9F 15	AD 9F 15
Missiles	\$52A2	8D B6 51	AD B6 51
Fireballs	\$5270	8D B4 51	AD B4 51
Nucr Missiles	\$52F9	CE B8 51	AD B8 51

I have a friend who is a teacher, and I have gotten a lot of software to crack from her. Here are some of the softkeys.

IIGS Softkey for...

World Geograph

MECC

Requirements:

2 Blank 3-1/2" disks
 Copy II+ version 8.0 or later or,
 any 3-1/2" copy program and sector editor.

World GeoGraph is a great world geography data base with excellent graphics. This program should be in all school software libraries. However, as with all MECC software, it is copy protected.

The copy protection scheme was of the same type used on Calendar Crafter. It only requires one byte to be changed.

1. Copy both disks, ignoring the bad block (\$0007) on disk 1.
2. With a sector editor, search for C9 27 00 F0 02 18 60 38 and change the 38 to 18 and save the block back to disk.

Hide the original in a safe place (away from students).

Softkey for...

Conquering Fractions; + and -.

Conquering Fractions; * and /.

Conquering Decimals + and -.

Conquering Decimals * and /.

Equation Math

Mystery Objects

Wood Car Rally

Coordinate Math

Fraction Practice Unlimited

Fraction Concepts

Decimal Concepts

MECC

Requirements:

1 Blank 5.25" disk for each program to be softkeyed
 COPYA
 Sector Editor

This method works on all MECC 5-1/4" disks that use ProDOS (at least the eleven disks that I tried).

1. Boot DOS 3.3
POKE 47397,24
POKE 47398,96
2. Insert a disk with COPYA.
RUN COPYA
3. Follow prompts and copy the disk(s).
4. With a sector editor, search for 90 03 46, then look 3 bytes further for a 60. In other words you're looking for 90 03 46 xx xx 60 where xx may be any byte.
5. Replace the six bytes with 18 EA EA EA EA 60 and write the sector back to the disk.
6. Replace (copy over) the MECC ProDOS file with your favorite ProDOS 8, like one with Birds Better Bye.

I found that if the first occurrence of 90 03 46 is not followed by xx xx 60 then 90 03 46 xx xx 60 will only occur once on the disk. If the first occurrence of 90 03 46 was followed by xx xx 60, then it would occur twice on the disk, and you would need to change them both!

Playing Tip for...

Bard's Tale III

Electronic Arts

To David Stewart: To change someone to a Geomancer: Do not kill Ferofist in Kinestia! If you do not fight him, he will show you the access to the Geomancer rite of passage.

To get the Rainbow Rose: In the Violet Mountain is the lair of the Rainbow Dragon. After you kill the dragon, you will see a pool of blood. Using a wineskin, fill it with the blood. Back in the wilderness of Luncencia at 3W, 2N there is a rose bush with no roses on it. Use the dragon's blood on the rose bush! Voila, a Rainbow Rose appears!

Sandy Lapp NY

Here is a softkey for two games on the Most Wanted list that use exactly the same protection. Using a nibble editor you will find that the normal epilogs have been changed from DE AA to FF EB. It seems to be a form of the Pro-Lok protection that a lot of companies are now using.

Softkey for...

LA Crackdown

Epyx

SOKO-BAN

Spectrum Holobyte

Requirements:

Apple IIe
DOS 3.3 System Disk
Sector Editor such as Copy II+

1. Boot DOS 3.3 system disk
2. Tell DOS to ignore checksum and epilog errors and use COPYA to copy the disk.

POKE 47426,24

RUN COPYA

3. Make sector edits to the copy you just made.

Trk	Sct	Byte(s)	From	To
00	05	0C-29	A9 0A 85 F6 A6 2B BD 89 C0 BD 8E C0 A9 80 85 FD C6 FD F0 74 20 A6 25 B0 6F A5 F9 C9 0F C0	A9 E7 85 F8 85 FB A9 FC 85 F9 85 FC 85 FF A9 EE 85 FA 85 FD 85 FE A6 2B BD 89 C0 18 90 52

If you don't have COPYA, Super IOB with FAST.CON can be used by inserting a line 1011, POKE 47426,24 into FAST.CON. and copying your disks.

Softkey for...

Mavis Beacon Teaches Typing

(IIe)

Software Toolworks

This was a tough little nut to crack simply because I was going about looking for the protection in the wrong way. The disk can be cataloged and the files read. I thought the protection was in one of the files. I was wrong. When I first did a nibble copy of the disk, it looked like it copied and even booted. But then the first screen on the copy instructed me to insert side B. When the original boots, you get a Menu and from there you can choose different options. The copy was worthless as it just put me in an endless loop. By checking the disk with Copy II+ sector nibble editor, I learned that the disk seemed normal except for track 0, sector 0. epilog bytes were AA DE instead of the standard DE AA. This seemed strange, but I really didn't know where to go from there. This disk really puzzled me. I could copy all the files onto another disk and add a DOS, but it still wouldn't work. On a hunch, I decided to investigate track 0, sector 0, a little closer.

Disassembled I got this starting at address E8:

20 B5 B7	JSR	\$B7B5
90 08	BCC	\$09F5
AD FF B7	LDA	\$B7FF
09 81	ORA	#\$81
8D FF B7	STA	\$B7FF
60	RTS	

I will not pretend to really understand the reason but I changed addresses EB and EC from 90 08 (branch if carry clear) to 18 60 which is clearing the carry and returning. Well it works anyway.

1. Boot DOS 3.3 System Disk.
2. Tell DOS to ignore checksum and epilog errors and use COPYA to copy the disk.

POKE 47426,24

RUN COPYA

3. Sector edit the copy with CopyII+ or sector editor of your choice. Change Track 0, Sector 0, bytes \$EB-EC from 90 08 to 18 60. Only the boot side of the disk is protected.

Playing Tip for...

Leisure Suit Larry (IIe)

Sierra OnLine

I hate to have to answer all those stupid questions each time you boot up the game. My sixteen year old son can whiz through them, but sometimes they give me a hard time. I found that when you get to the first question, press ctrl X and it will pass up the questions and let you play

the game. I tried this after reading one of your IBM articles whereas they used option X (I believe) to pass up the questions.

Also, I own a Franklin Ace 1200 (Apple II+ compatible) besides an Apple IIe. I have a hunch there are still a few Franklin Computers going strong out there somewhere. (They're solid workhorses.)

Here is a tip on fixing ProDOS to work on a Franklin. With a sector editor, search for the bytes (69 0B D0 03) usually found on track one. Change D0 03 to EA EA and it should work.

ProDOS 8 v 1.2 changes

Trk	Sct	Byte(s)	From	To
01	07	0C-0D	D0 03	EA EA

ProDOS 8 v 1.4 changes

Trk	Sct	Byte(s)	From	To
01	07	2B-2C	D0 03	EA EA

Be sure to write the changes back to the disk.

Ⓢ I now would like to ask a favor. If anyone has cracked or even copied Dark Lord by Datasoft, please forward the information to Computist. My cat Lucy has mutilated the jacket on my original. Any help would be appreciated.

Mike Egnotovich PA

Softkey for...

Where in the USA is Carmen

San Diego

Broderbund

Requirements:

COPYA
Four Blank Disk Sides
Copy II+ v. 8.x (or another ProDOS disk/file copier)

Bag of Tricks (Optional)

Locksmith 6/7.0 FastCopy

Where in the USA Is Carmen San Diego? (Original 3 disk series)

First of all, when I was presented with these disks by a co-worker, I thought no sweat! I've seen Softkeys for those programs in past issues of COMPUTIST (Issues #59 and #64). Well I was wrong on both counts. Those issues contained Bit Copy hints, not softkeys and therefore, there was some sweat involved (just a little, though).

Experience has shown, that even though you get a softkey/bit copy from someone else, many software companies vary the protection they use from release to release of the same title. That's why, regardless of the reliability of the source of deprotection hints, you are always better in the long run to develop and use a standard methodology for deprotecting, and treat those "hints" as just that.

My "standard" methodology (which I vary slightly from time to time - i.e., I "cheat") involves first trying to copy a disk with a standard copier (Locksmith FastCopy or Copy II+ Disk Copy). In this case Locksmith FastCopy was unable to read the "A" (Side 1) disk. This was indicative of some kind of non-standard formatting. My next step should have been to use Bag of Tricks (TRAX mode) to see what format alterations were present. But, as I warned above (and also indicated that I am sometimes tempted to "cheat"), I thought I'd circumvent the whole process and try the Bit Copy hints in the back issues of COMPUTIST. Well, to no-one's surprise but mine, the Bit Copy hints provided by Messrs. Hart and Kieth did not work on this disk. The Auto Bit Copy of Copy II+ using the parameters on versions 8.2 through 8.4 produced many 2 and 5 error codes. Back to the drawing boards. My old trusty Bag of Tricks - TRAX, came to the rescue as it has many times before.

TRAX showed that the "A" disk contained altered Epilog bytes for both addresses and data. No problem here. This type of protection is frequently encountered as a first stage protection. Its removal is simple.

1. Boot up with normal DOS 3.3.
2. At the BASIC prompt (>) enter the following:
POKE 47426,24

This poke tells COPYA to ignore Checksums and Epilogs when copying (OK, Kay Jun. Do you see now why you did this for your Microzine disks? (re: COMPUTIST #66, page 12). The developer obviously had altered either normal DOS Checksum routines or used non-standard epilogs. As DOS expects hex values DE AA for epilogs to denote data/instruction demarcations on disk sectors, with altered epilogs, it can't tell where the data/instructions end. The only way to read these disks is to read them with an altered DOS that knows what the actual epilog values are or, using the beauty of COPYA, to ignore them completely, "normalizing the resultant copy).

3. Run COPYA using Carmen USA master side
A as the source and one of your blank disks as the target.
RUN COPYA

Bingo! What you wind up with is a ProDOS disk that can be cataloged by Copy II+. This disk will run from a program selector like Squirt, ProSel, etc. but it won't boot by itself. Now when you catalog it with Copy II+, you'll notice that

you'll see a pathname /CARMEN.USA with a bunch of files but the Block count (Used, Remaining and Total) that you see at the bottom of the screen is out of whack (the numbers don't add up to a 280 block disk). This is indicative of an index problem. However, the files in the index seem normal.

4. Format a blank disk using Copy II+ and name it /A (anything other than /CARMEN.USA will do). Copy all of the files from the copy to the newly formatted disk and renamed the new disk /CARMEN.USA.

That's all there is. The new disk boots fine. The nibble checks that were alluded to in the earlier COMPUTIST articles were not present on this version.

The other two sides to the original, "B" and "C", were not copy protected. They can be copied using Locksmith FastCopy or Copy II+ Disk Copy.

This is a new version of Carmen USA. The files were as new as April 1989 and ProDOS version 1.7 was on the originals. The deprotected copy can be launched from a program selector or probably installed on a 3.5" disk or hard disk. Be sure to name the subdirectory where it is installed /CARMEN.USA as the USA.SYSTEM system file checks for that name.

To Michael Reese: (re: COMPUTIST #66, page 14.) I hope the editors comments helped. I can add no more information. The Patchwork/Negotiation Edge/Management Edge softkey worked on my //e and //c machines, with disk controller in slot 6.

Bill Jetzer WI

Ⓢ I read the "AppleEar" article from Issue #10. Since I have a //c, there is no cassette jack. I was wondering if it is possible to hook up a cassette to the joystick port to input the data required by the program.

I just received AppleWorks 3.0. This is a major improvement over older versions. One improvement that is never mentioned in the ads is the fact that a version of Bird's Better Bye has been installed on the ProDOS 8 v1.7 file.

I was confused when it came time to print. The printer drivers aren't reliable. I have an Imagewriter II (IW2), so I added it to my list. I then tested out all the cpi, boldface, superscript, etc options. I printed it with the IW1 driver and everything went fine. But when I printed the document to the IW2, most of the lines were indented several spaces. Then I removed the IW2 and added another IW2, and again printed the document. This time, the IW2 printed correctly, but the IW1 had problems similar to the IW2, only not as severe. We can probably expect a version 3.1 in the near future.

On the Startup disk, there is a program called FASTCOPY. You can use it to copy your disks and compare two disks in two passes (on a 128k machine). However, it requires that the original be write protected. To get around that, follow these instructions:
BLOAD FASTCOPY,AS\$2000,TSYS
CALL-151
5F52:1 (from 0)
BSAVE FASTCOPY,AS\$2000,TSYS

You can also format disks with FASTCOPY. It isn't an option, but if you try to copy onto an unformatted disk, it will allow you to choose whether or not to format the disk.

I heard that BI version 1.3 has been released and has a major bug. Apparently, someone forgot a branch in the source code, so instead of giving you a NO BUFFERS AVAILABLE message, it falls through the BSAVE code! I don't have that version, but if you do, either use an earlier version or wait for 1.4.

Ⓢ How about keeping us up to date on current versions of ProDOS and the BI? They release new versions left and right, so it's hard to keep track.

Zeno Kopecky IL

Ⓢ Help! I need info on how to copy Blade at Blackpool (Sirius) and Escape (Bantam/Firebird).

James Hodge CT

Softkey for...

Balance of Power 1990 v. 2.08

Chris Crawford

This is a really ambitious game. Its main drawback is its lack of speed. Running it from a RAM disk or a hard disk with a TransWarp GS would be a big help. It requires a megabyte of free memory, which seems excessive.

For some reason the game designers felt that password protection was necessary. I don't know why, because the game also requires the manual if you're going to play it effectively, and if you've got the manual, you've got the passwords.

Oh, well, removing the password protection is probably easier than playing the game.

To find where to patch the code, I went into "VisitMonitor" and searched for the string "ARE

YOUR PAPERS IN ORDER". When I found that, I searched for a reference to the address of the string. When I found THAT, I knew I was in the password routine. Then it was just a matter of identifying tool calls and address references and figuring out what was going on.

The list of passwords can be found in bank \$F memory when the game is at the point of asking for your papers. They usually occur between F/2700.3500 and/or F/6980.7180.

The file that gets patched is "BOPII.SYS16" and it has a MOD Date of 23-FEB-89, 1:04, and a CREATE Date of 19-FEB-89, 15:46. The following edits will do it:

Blk	Byte(s)	From	To
\$5F6	1A9	F0 03	EA EA

Softkey Update for...

Chessmaster 2100 v1.1

Software Toolworks

Issue #66 of Computist carried an article by me detailing the removal of the Chessmaster ver. 1.01 password protection. Well, Software Toolworks came out with ver. 1.1, and the old patches don't quite fit. This was pointed out to me by Bob Cherochak, who wondered what was needed to fix the new release.

I was able to retrace the steps I followed originally and come up with a new patch that is an improvement over the old one. If you have Chessmaster 2100, ver. 1.1 or 1.01, only one byte needs to be changed:

Version 1.1

Blk	Byte(s)	From	To	
B1k	304	195	1B	02

Version 1.01

Blk	Byte(s)	From	To	
B1k	4E4	1D6	1B	02

This changes the length of a branch in the password routine so it skips to the end of the routine but still allows it to set up some "registers and locations correctly.

The following piece of code is from Chessmaster ver. 1.1, but it is, with the exception of some addresses, identical to the routine used in ver. 1.01. If future releases of Chessmaster continue to use <the same routine it should be simple to scan a disk for the appropriate byte pattern. I'd recommend looking for the 5 bytes starting at 3/8DB8, and then checking to see if the 3 bytes at 3/8DC8 (82 1B 02) are where they should be, relative to the other bytes.

8D9D:A2 02 2B	LDX	#\$2B02
8DA0:22 00 00	E1JSL	SEI\$0000
8DA4:80 03	BCS	8DA9
8DA6:A9 00 00	LDA	#\$0000
8DA9:8F 02 08 01	STA	\$010802
8DAD:AE C0 09	LDX	\$09C0
8DB0:AD BE 09	LDA	\$09BE
8DB3:DA	PHX	
8DB4:48	PHA	
8DB5:F4 78 00	PEA	\$0078
8DB8:D4 9C	PEI	\$9C
8DBA:D4 9A	PEI	\$9A
8DBC:22 9A 8B 03	JSL	\$038B9A
8DC0:7A	PLY	
8DC1:7A	PLY	
8DC2:7A	PLY	
8DC3:7A	PLY	
8DC4:7A	PLY	
8DC5:AD BC 09	LDA	\$09BC
8DC8:82 1B 02	BRL	\$8FE6 this gets changed to BRL 8FC0 (1B -> 02)
8DCB:F4 00 00	PEA	\$0000
8DCE:7B	TDC	
8DCF:18	CLC	
8DD0:69 D8 00	ADC	#\$00D8
8DD3:48	PHA	
8DD4:F4 00 00	PEA	\$0000
8DD7:7B	TDC	
8DD8:18	CLC	
8DD9:69 CA 00	ADC	#\$00CA
8DDC:48	PHA	
8DDD:F4 00 00	PEA	\$0000
8DE0:7B	TDC	

Softkey for...

World GeoGraph v1.0

MECC

To produce a broken version of World Geograph perform the following block edit:

Blk	Byte(s)	From	To
\$6B	173	B0 26	80 20
	195	00 D0 02	A9 27 00

The protection code is shown below. The program reads block 9, compares the first 18 bytes read, then expects an I/O error (I/O error) when it tries to read block 8. The above edits change the "BCS B1B5" to "BRA B1AF" and put a "LDA \$S0027" at \$B1AF, just ahead of the CLC instruction. This seems to satisfy the protection requirements.

If the following code looks vaguely familiar, that's because it is protection used by MECC on version 1.1 of their Calendar Crafter program. The routine on World GeoGraph is virtually identical, differing only in its location on disk and in memory, and the number of bytes they compare.

The file that gets patched is "WORLD.GEOGRAPH", with a MOD Date of 12-DEC-

88, 11:45, and a CREATE Date of 10-NOV-88, 19:13.

B17D:A9 09 00 LDA #0009
B180:8D BC B1 STA B1BC Block num.
B183:22 A8 00 E1 JSL E100A8 Read block 9
B187:22 00 0022
B189:B6 B1 01 00 0001B1B6
B18D:B0 26 BCS B1B5 shouldn't branch
B18F:A2 13 00 LDX #0011
B192:BD C0 B3 LDA B3C0,X
B195:DD C0 B1 CMP B1C0,X check what was read
B198:D0 1A BNE B1B4 shouldn't branch
B19A:CA DEX
B19B:CA DEX
B19C:10 F4 BPL B192
B19E:CE BC B1 DEC B1BC set to read blk 8 (bad block)

B1A1:22 A8 00 E1 JSL E100A8 try to read it
B1A5:22 00 0022
B1A7:B6 B1 01 00 0001B1B6
B1AB:90 07 BCC B1B4 shouldn't branch
B1AD:C9 27 00 CMP #0027 make sure it was I/O error
B1B0:D0 02 BNE B1B4 if not, set carry, return
B1B2:18 CLC
B1B3:60 RTS
B1B4:38 SEC
B1B5:60 RTS

Softkey for...

King of Chicago
Cinemaware

King of Chicago is an "interactive movie" about a punk called Pinky Callahan trying to take over the mob action in 1930's Chicago. Actually, it reminds me of an adventure game, but with great graphics.

The game has two major problems: 1) the operating system on the disk is GS/OS but the disk is in a 4:1 interleave format, and 2) the disk uses a nibble count on tracks \$20 and \$21. To correct the interleave problem means removing the protection. Two minor problems the game suffers from: too much disk swapping and too great a memory requirement. If you want to copy the main files from /KING1 to a ramdisk (to avoid disk swapping) you'll need a 576K ramdisk, and you'll still need an additional 768K of free ram to run the game. Two 3.5 disk drives, a hard disk or at least 1.25 megabytes of additional memory (for a ram disk) should make the game less of a hassle to use.

A utility called 2gs Internals, from the Merlin 16+ disk, gave me some idea about where to look. A file called KINGSHELL.SYS16 was using code in bank 0, and that seemed suspicious. Tracing the call to the routine at 0/8000 took me to some code in bank 1. When I tried to skip around the call to the code in bank 0 the game seemed to work for awhile, but then it crashed. Ohhh, nuts! Back to the drawing board.

I decided to figure out what was going on in bank 0, so I used Sourceror (from Merlin 8/16) to completely disassemble the protection code. I found calls to the Smartport to position the drive head, and the following code (at 0/82A7) to read and count the nibbles on a track. The Smartport calls don't show up on disk because they are created in memory at execution time. The commands that actually read the nibbles are sneaky, because they use direct (zero) page indirect addressing (the "LDA (\$F0)" instructions).

I decided to alter the nibble reader routine to give the program the values it wanted. I booted the original disk, and when it reached the point where it started its protection routine I visited the monitor and substituted a BRK for the RTS at the end of the nibble reader. When the game crashed, I entered "ctrl T <ret>" to clear the super hi-res screen and view the values in the X and Y registers. Then I inserted the code needed to set the registers and locations \$F4 and \$F5 properly. The first patch seemed to work, but I had noticed when I was tracing code that the protection routine was called twice. I rearranged the patch to insert a BRK in the code for the second pass, and I found a different set of values are produced by the nibble count on track \$21. The final patch looks like this:

LDX #A0 x and y values for the 1st pass
LDY #20
LDA #4D x value for 2nd pass
STA 0082AC modify LDX instruction
LDA #1E y value for 2nd pass
STA 0082AE modify LDY instruction
BRA 82EF skip past nibble count

The patch I came up with is as follows:

Blk Byte From To
570 148 A9 00 85 F4 85 F5 A2 A0 A0 20 A9 4D
A2 00 A0 F8 84 F7 8F AC 82 00 A9 1E
E8 D0 04 E6 F7 F0 8F AE 82 00 80 32
570 193 A6 F4 A4 86 F4 84

The first part of the patch (at \$82AB in memory) loads X and Y with \$A0 and \$20, respectively, alters the values that will be loaded into X and Y on the second pass, and then branches to \$82EF, where the second part of the patch has converted the LDX and LDY instructions to STX and STY. This produces a game that seems to work without crashing.

The disks /KING1 and /KING2 can be set up for faster booting and reading. Format a couple of disks in 2:1 interleave and then fast copy the originals to the "2:1" disks. Apply the above

patches to /KING1 and you're ready to go. Boot time is reduced by approximately 25 seconds. I tried adding a /DESK.ACCS subdirectory and the GS/OS disk cache but it didn't seem to improve disk access.

The best arrangement I've found for my system (2gs, 01 ROMs, AE Ramkeeper, GS Ram with 1.25 meg, 1 3.5 Apple drive) is to set up a 575K ROM disk, copy KING.1 and KINGSHELL.SYS16 to the ROM disk, boot on a GS/OS system disk, and execute the game from the ROM disk, but with the /KING1 disk in the drive. After the game starts you can put the /KING2 disk in the drive and no further disk handling will be needed.

Incidentally, you can tell whether a disk is 2:1 or 4:1 interleave by using the Copy 2+ "Verify Disk" option. 2:1 disks verify in about 245 seconds, while 4:1 disks take about 90 seconds.

The file affected by the above patches is KINGSHELL.SYS16 with a create date of 18-AUG-88, 12:88, and a modify date of 10-OCT-88, 17:06.

82A3:A9 30 LDA #30
82A5:38 SEC
82A6:60 RTS
82A7:85 F0 STA \$F0 store #SEC
82A9:84 F1 STY \$F1 store #SC0
82AB:A9 00 LDA #300
82AD:85 F4 STA \$F4 zero key locations
82AF:85 F5 STA \$F5
82B1:A2 00 LDX #300
82B3:A0 F8 LDY #300
82B5:84 F7 STY \$F7 counter
82B7:E8 INX
82B8:D0 04 BNE 82BE
82BA:E6 F7 INC \$F7
82BC:F0 E5 BEQ 82A3 error exit
82BE:B2 F0 LDA (\$F0) nibble reading!
82C0:10 FC BPL 82BE
82C2:C9 D5 CMP #3D5 find D5/AA/96
82C4:D0 F1 BNE 82B7
82C6:B2 F0 LDA (\$F0)
82C8:10 FC BPL 82C6
82CA:C9 AA CMP #3AA
82CC:D0 F4 BNE 82C2
82CE:B2 F0 LDA (\$F0)
82D0:10 FC BPL 82CE
82D2:C9 96 CMP #396
82D4:D0 EC BNE 82C2
82D6:18 CLC
82D7:8A TXA
82D8:89 05 ADC #305
82DA:AA TAX
82DB:A5 F7 LDA \$F7
82DD:89 08 ADC #308
82DF:A8 TAY
82E0:18 CLC
82E1:8A TXA
82E2:85 F4 ADC \$F4
82E4:85 F4 STA \$F4
82E6:98 TYA
82E7:85 F5 ADC \$F5
82E9:85 F5 STA \$F5
82EB:C6 F8 DEC \$F8 counter - orig. \$0A
82ED:D0 C2 BNE 82B1 branch back 9 times
82EF:A9 00 LDA #300
82F1:A6 F4 LDX \$F4
82F3:A4 F5 LDY \$F5
82F5:18 CLC
82F6:60 RTS

Softkey for...

Star Trek
Sega

Note: I did this work several years ago on an Apple 2e. This version of Star Trek (how many versions are there? Dozens, probably.) is by Sega. They call it a strategic operations simulator, but it looks like an arcade game to me. Oh, well, it has reasonably nice graphics, and it is not too hard to make a BRUNable file of. For the technically minded, I will describe the methods used to "break" Star Trek. If the reader is interested only in backing up their disk, you can skip the next few paragraphs.

Personally, I find "the thrill of the hunt is 90% in the chase". The capture of the quarry is anticlimactic. Star Trek's protection scheme was probably a better game than the game itself.

For starters, part of the protection seems to rely on some "undefined" 6502 opcodes. They are instructions that the 6502 will recognize, but that are not documented, and will not disassemble properly. (Because of this, the game has problems booting on a 2gs. The game keeps re-booting itself. Hitting control reset seems to cause the game to load itself and execute.) They are used on T0, S0, and they make it difficult to trace the code. Most of T0, S0 looks like garbage, anyway.

Upon booting, T0, S0 "un-scrambles" it's data as it copies it's first \$98 bytes to \$700, part of the text screen buffer. After that, control jumps to \$705, where the reset vector is set to \$200, and some code is copied to \$200. Two more sectors (sectors 1 and 2 on track 0) are read into the text screen area, and control is then passed to a routine at \$484. Once again, the reset vector is set to \$200, and some code is copied to \$200. The hi-res is switched on, some zero page locations are set up, and two bytes are pushed (very innocently) onto the stack. They will be used for a "funny jump" to \$7FD after the disk is read in. Finally, the stage two boot loops between 2

routines, one at \$400 and one at \$500, that position the drive arm and read a track, respectively.

The track read routine is interesting. First, it looks for a 5 byte header composed of the bytes \$B5, \$A7, \$E9, \$AB, and \$AB. The data is stored in "4 and 4" encoded form, as opposed to the normal DOS "6 and 2" form. It takes 512 bytes of disk data to make 256 bytes of "real" data (DOS 3.3 needs 342 bytes of disk data to make a "256 byte" sector). There are 12 "sectors" worth of info on a track, and 12 tracks are used. After the program is read in, a return is executed, producing the "funny jump" I mentioned earlier, and starting the game.

I was able to get the information from the screen buffer area by using the XFER.BOOT routine from the article "Secret Weapon: Ramcard", by Ken Greenlaw (Hardcore COMPUTIST #16). I booted Star Trek into the auxiliary memory of my //e, and hit control reset while the disk was still loading. When the game starts it changes the data in the screen area, and all the useful information gets stepped on. The advantage of booting into auxiliary memory is that a reset won't blow away the auxiliary screen.

Using the information from the read routine, it is easy to supply parms to a bit copier to make a valid copy-protected backup. In COPY 2+, use the manual sector copy option to copy track 0. Then, go into the manual bit copy routine and tell it to copy tracks 1 through \$C, without maintaining track synchronization or length. Lastly, there are 7 parms that need to be changed.

Parm Set to
A 5
E B5
F A7
10 E9
11 AB
12 AB
55 30

After setting the parms to the above values, COPY 2+ will copy Star Trek with no problems.

I edited T0, S1 of the copy so it would jump into the monitor instead of starting the game. If this idea appeals to you, edit bytes E0, E1 and E2 to the values 4C, 69 and FF, respectively. If you don't want to edit your original (or can't) don't worry, because the technique presented below will produce the same effect without altering the disk

The following code is used in conjunction with the code from the disk controller. The code can be created in place with a mini-assembler, or in a regular assembler. If you don't have access to an assembler, the hex values for the code can be entered in the monitor. However you go about getting the code, I would recommend saving the code to disk with the command: BSAVE GET.STR.TRK, A\$66F8, L\$34. The next step is to move the disk controller code down to \$6600 with the command: 6600<C600.C6F7M. The GET.STR.TRK code (GTS) replaces the disk controller's jump to \$801. To boot Star Trek, issue the command: 6600G <ret>.

The GTS code is fairly simple. Once T0, S0 is booted in, the code from \$66F8 to \$6706 alters the address that the code at \$8B3 jumps to. It causes it to jump to \$6707, instead of \$705. The code at \$6707 through \$6715 sets the code at \$76C to jump back to \$6716, where the final patch is installed at \$4E0. The final patch will cause a jump into the monitor, instead of starting the game. The patches have to be installed in this fashion if a would be Captain Kirk doesn't want to sector edit his/her disk, because the stage 1 boot has a couple of steps.

ORG \$66F8
66F8:48 PHA
66F9:A9 07 LDA #307
66FB:8D B4 08 STA \$8B4
66FE:A9 67 LDA #367
6700:8D B5 08 STA \$8B5
6703:68 PLA
6704:4C 01 08 JMP \$801
6707:48 PHA
6708:A9 16 LDA #316
670A:8D 6D 07 STA \$76D
670D:A9 67 LDA #367
670F:8D 6E 07 STA \$76E
6712:68 PLA
6713:4C 05 07 JMP \$705
6716:48 PHA
6717:A9 4C LDA #34C
6719:8D E0 04 STA \$4E0
671C:A9 69 LDA #369
671E:8D E1 04 STA \$4E1
6721:A9 FF LDA #3FF
6723:8D E2 04 STA \$4E2
6726:68 PLA
6727:4C 84 04 JMP \$0484

Booting the disk will load the complete Star Trek program and leave you in the monitor, starting at a screen full of hi-res junk. Type: C054

C051

And the screen returns to text page 1. At \$7FD is a JMP \$2400. At \$2400 is code to re-arrange data and code in memory. The monitor MOVE routine is used three times. The moves would look like this: 400<800.23FFM, 9000<6000.8FFFM, and 6000<3000.5FFFM. After the moves are done control is then passed

to \$7F0. It's obvious that everything needed for Star Trek is located between \$7FD and \$9000.

Hopefully, you have an initialized slave disk that won't load a HELLO program. At this point, you want to move the data from page 7-9 out of the way of a disk boot. From the monitor (you didn't leave, did you?) issue the command: 90FD<7FD.9FFM

Now, boot your HELLOless slave disk 6 ctrl P

Get back into the monitor and restore the code previously moved to it's original location.

CALL-151
7FD<90FD.92FFM

DOS needs a patch to BSAVE the Star Trek program. The command to patch the allowable length for a BSAVE is:

A964:FF

All that's left (finally!) is to: BSAVE STR.TRK, A\$7FD, L\$8804

The patch made to DOS is temporary, unless you init a disk with it. It is only needed for bsaving files greater than \$7FFF (32767 decimal) in length. Extra long files will still load correctly with an un-patched DOS.

A final note: if you don't like programs that cause reset to reboot, change the value at \$3D33 to \$69, and the value at \$3D38 to \$FF, and then reset will send the Enterprise crashing into the monitor (sounds serious, doesn't it?). If you would rather end up in BASIC, use \$00 and \$E0 instead of \$69 and \$FF

Softkey for...

Gnarly Golf (GS)

Fanfanel Britannica Software

This is a strange game at two levels; at the game level, and at the software level.

The game uses ProDOS 8 v.1.6 but it is generally 16 bit code, so it can be difficult to follow when it switches register widths. While there are subroutines, the code uses a lot of branch instructions, and that means that the monitor Pattern search command may not be able to locate address references. Oh, well, I was still able to figure out what they were doing.

The protection has two parts. The first part is a bad block check. The second part examines the results of the first part. Get past both checks and you're in the game. The code for the block read and comparison is shown below.

I came up with two patches. One is quick and simple, but it may not set up memory properly for some other part of the game. It seems to work, though. The second patch is only slightly more involved, but I think it is a better solution. Only one of the patches needs to be used. I leave the choice up to the individual.

Blk Byte(s) From To
\$166 \$D8 F0 DE 02 DF

This patch changes the "BRL \$1FA3" at \$40B0 to "BRL \$1FB5".

The more complex patch disables the bad block check at \$2701 and changes the "CMP \$5000,Y" at \$40A2 to "STA \$5000,Y". NOPping the readblock command in block \$159 allows the game to start a little faster and with less noise, because the drive head isn't being run to the inside track on the disk. The patch to block \$166 should leave memory in the same state as the original game.

Blk Byte(s) From To
\$159 \$127 20 00 BF 80 EA EA EA EA
10 27 EA EA
\$166 \$C9 D9 99
\$CC D0 08 EA EA

26FF:38 SEC
2700:FB XCE set emulation mode
2701:20 00 BF JSR \$BF00 MLI read block \$63F
2704:80 HEX 80
2705:10 27 DA \$2710
2707:08 PHP
2708:18 CLC
2709:FB XCE back to native mode
270A:28 PLP
270B:C2 30 REP \$30
270D:82 1D 12 BRL \$392D
2710:03 50 00 50 HEX 035000503F06
2714:3F 06

4091:B0 1D BCS \$40B0
4093:C2 30 REP \$30
4095:A0 FE 01 LDY #301FE
4098:A2 00 00 LDX #30000
409B:98 TYA
409C:18 CLC
409D:E8 INX
409E:8A TXA generate test value
409F:2A ROL
40A0:AA TAX
40A1:08 PHP
40A2:D9 00 50 CMP \$5000,Y compare to disk values
40A5:D0 08 BNE \$40AF fail, retry
40A7:28 PLP
40A8:88 DEY
40A9:88 DEY
40AA:10 F1 BPL \$409D
40AC:82 06 DF BRL \$1FB5 pass, start game
40AF:28 PLP
40B0:82 F0 DE BRL \$1FA3 fail, retry

Where in the USA is Carmen Sandiego v2.00

Broderbund

This game is the version on a 3.5 inch disk, designed for the 2c and 2e. The graphics look similar to the older "Carmen", but the disk is ProDOS, and the protection is a nibble count on tracks \$20 and \$21, just like "Where in the WORLD is Carmen Sandiego". The protection code is contained in a file called "Z". When you boot the disk, or "-USA.SYSTEM", Z is loaded at \$4000 and executed by a JSR \$4000. The logic in Z is quite similar to the logic in "King of Chicago" and "Carmen World".

I came up with 3 different patches, ranging from simple to involved. Only one patch needs to be applied. The first two patches will work with 2e's, 2c's, and 2gs, while the last patch is 2gs specific.

The simple patch may, or may not, be "correct". I know that a patch like the following won't work for "King of Chicago" because it prevents certain locations from being initialized properly. So, the "simple" patch might not work in the long run on "Carmen.USA", but you may want to try it and test it. It DOES allow you to start a game, but I haven't tested it beyond that point. I have more faith in the more complex patches.

The simple patch:

Blk	Byte(s)	From	To
\$27	\$67	20 98 21	4C A2 20

The simple patch instructs "USA.SYSTEM" to skip loading and executing "Z".

The second patch, also applied to "USA.SYSTEM", allows "Z" to load and execute, but it instructs "USA.SYSTEM" to ignore the returned values.

Blk	Byte(s)	From	To
\$27	\$77	B0 E8	80 0E
	\$92	B0 CD	80 0E

Last, there is the complex patch, a 2gs specific (or, more accurately, 65802/65816 specific) patch. The code for the patch follows:

43DE:A2 48	LDX	#\$48
43E0:A0 1E	LDY	#\$1E
43E2:A9 9F	LDA	#\$9F
43E4:8F DF 43 00	STA	\$0043DF
43E8:A9 20	LDA	#\$20
43EA:8F E1 43 00	STA	\$0043E1
43EE:A9 76	LDA	#\$76
43F0:48	PHA	
43F1:28	PLP	
43F2:A9 00	LDA	#\$00
43F4:18	CLC	
43F5:60	RTS	

I can guarantee that 2e's/2c's will not work with the 2gs patch. There is probably some softswitch that needs to be flipped somewhere, but I don't know which one and I'm not interested in spending hours playing with them. To adapt this for the 2e/2c the patch would need to have a softswitch flipped since it can't use the 65816 specific instructions that include bank references. I have tried flipping the softswitches at \$C004 and \$C054, but that didn't work. If someone works it out, be sure to let Computist know.

To summarize the patch:

Blk	Byte(s)	From	To
\$D7	\$148	8D	60
	\$1DE	A2 32 A0 00	A2 48 A0 1E
		AD EE C0 10	A9 9F 8F DF
		FB C9 D5 F0	43 00 A9 20
		06 EB D0 F4	8F E1 43 00
		C8 D0 F1 AD	A9 76 48 28
		EE C0 10 FB	A9 00 18 60

The nibble count routine (the code that gets the above patch) is different in some details to the nibble counter in the other programs mentioned, but the purpose is obvious. It's shown below. The first part of the complex patch, replacing \$8D with \$60, prevents the nibble counter from being modified.

The files "USA.SYSTEM" and "Z" both have Create dates of 25-OCT-04 at 14:17. The Modified date on "USA.SYSTEM" is 26-APR-89, 18:47, and on "Z" the date is 24-OCT-88, 07:05.

43DE:A2 32	LDX	#\$32
43E0:A0 00	LDY	#\$00
43E2:AD EE C0	LDA	\$C0EE this address is self-modified
43E5:10 FB	BPL	\$43E2
43E7:C9 D5	CMP	#\$D5
43E9:F0 06	BEQ	\$43F1 find D5/AA/96
43EB:E8	INX	build nibble count in X and Y
43EC:D0 F4	BNE	\$43E2
43EE:C8	INY	
43EF:D0 F1	BNE	\$43E2
43F1:AD EE C0	LDA	\$C0EE this address is self-modified
43F4:10 FB	BPL	\$43F1
43F6:C9 AA	CMP	#\$AA
43F8:D0 F1	BNE	\$43EB
43FA:AD EE C0	LDA	\$C0EE this address is self-modified
43FD:10 FB	BPL	\$43FA
43FF:C9 96	CMP	#\$96
4401:D0 E8	BNE	\$43EB
4403:C6 E5	DEC	\$E5
4405:D0 DB	BNE	\$43E2

4407:A9 00	LDA	#\$00
4409:18	CLC	
440A:60	RTS	

Softkey for...

Where in the World is Carmen Sandiego (GS v1.0)

Broderbund

I started working on this game after spending some time breaking the protection on The King of Chicago. I was both pleased and amazed to find that the protection on "Carmen" was virtually identical to "King". Some days you get lucky, I guess.

The nibble reading routine was at the same address as in "King" and it was byte-for-byte identical. The calling routines, although at different addresses, were also similar.

Using the monitor's Pattern search and List commands to track back through the calling routines produced the following "trail" - a routine at 1/182D contained 2 calls to 1/1877. That routine called 1/A376, and that routine called 0/8000. To find the values the program wanted from the nibble reader was easy. I launched CARMEN.WORLD from a copy of the original disk, and when it asked for the original I got into the monitor (through Visit Monitor) and installed a "BRK" (hex 00) after the first call to 1/1877. Then I put the original disk in the drive and let the program continue. When it crashed, I entered "ctrl T <ret>" to switch off the super hi-res and noted the values in the X and Y registers. I repeated the process but put a "BRK" after the second call to 1/1877 to get the second set of values. The code for the patch is nearly identical to the patch for "King" with the exception of the values for the X register. The code for the patch looks like this:

LDX #B2	x and y values for the 1st pass
LDY #20	
LDA #5D	x value for 2nd pass
STA 0082AC	modify LDX instruction
LDA #1E	y value for 2nd pass
STA 0082AE	modify LDY instruction
BRA 82EF	skip past nibble count

The patch is as follows:

Blk	Byte	From	To
\$4F \$125	A9 00 85 F4 85 F5	A2 B2 A0 20 A9 5D	
	A2 00 A0 F8 84 F7	8F AC 82 00 A9 1E	
	E8 D0 04 E6 F7 F0	8F AE 82 00 80 32	
\$4F \$16B	A6 F4 A4	86 F4 84	

The first part of the patch (at \$82AB in memory) loads X and Y with \$B2 and \$20, respectively, alters the values that will be loaded into X and Y on the second pass, and then branches to \$82EF, where the second part of the patch has converted the LDX and LDY instructions to STX and STY. This produces a game that seems to work.

It appears that only disk 2 is used for the nibble count. Disk 1 is just a 3.2 system disk. While the documentation says that "Carmen" needs that particular system, I found that it could be launched from GS/OS system 4.0, but not system 5.0. If you use GS/OS you'll want to copy disk 2 to a 2:1 interleave formatted disk. If you want to install a disk cache on disk 1 you can make room by deleting all the Appletalk related files in the /Appletalk and /System.setup subdirectories, as well as the various drivers in /Drivers, since "Carmen" doesn't need them.

The patches are for the file CARMEN.WORLD, with Create and Modify dates of 06-JAN-89 at 18:53, on the disk /CARMEN.WORLD2.

Ro France

Beginners Guide to "PACMAN" deprotection.

Softkey for...

Pacman

Thunder Mountain

Requirements:

Apple II
a blank formatted DOS 3.3 disk
Super IOB
a way into the monitor
a nibble editor

I don't think it is necessary to present the game, it had its days of glory in the beginning 80's. Today it really shows its age, and even the protection is old-fashioned. Imagine what it could be, if it gets the treatment as "BrickOut" compared to Taito's "Arkanoid". Enough day-dreaming, let's get started.

Before you do anything, WRITE PROTECT the original. Done?? Boot Pacman, it is long to load, but no Woody Woodpecker noise of the read head indicating a nibble count. Just after the "Please Wait" and "Loading" messages you see for a moment the J prompt. It's a good sign that DOS is quite normal.

The next logical step is to try to COPY A it, but even with the Error Check disabled (B942:18 or POKE 47426,24) you get the "Unable To Read" message. This means they modified the Address

and/or Data markers. A quick look with the nibble editor confirms it, the Address prologue is something like DC 96 AB on some sectors, DD 96 AB on others; the epilogues are 96 FF. The Data field markers are modified to DD 96 EF and 96 FE.

Now let's check if there are other modifications. Boot your sector editor and customize the DOS 3.3 address and data markers to before mentioned values. This will permit access of some sectors but the code written means nothing, it's only garbaged data. Conclusion they modified not only the Markers, but the Translate Tables in the RWTS, too.

Before we decide the way to follow, let us check if Super IOB and the Swap Controller can do the work for us. (I'm unable to write a controller, having never tried to program in Applesoft BASIC).

Just in case prepare a fresh slave disk; boot your DOS Master and INIT the disk. Put it aside for the moment.

Now boot your original Pacman and at the J prompt break into the monitor (see page 4 of your favorite mag for details). Will we find the RWTS in its standard location? Check by typing B800L, it begins with:

B800-A2 00	LDX	#\$00
B802-A0 02	LDY	#\$02
B804-88	DEY	

That is the normal beginning of the RWTS under DOS 3.3. Bring the RWTS down to a safe place so we can BSAVE it.

1900<B800.BFFFFM

Boot your slave disk by typing from the monitor:
C600G

And at the prompt save the PACMAN RWTS.
BSAVE RWTS.PACMAN, A\$1900, L\$800

Save it here in case of a power failure or other incidents, otherwise you have to go through the whole process again. We will do a complete verification of the RWTS. It's the hard way, but you never know the programmers who create protections have a strange sense of humor. We will go through all 8 pages of the code and compare it to the normal DOS RWTS.

OK?, You got your listing ready, so here we go:

BLOOD RWTS.PACMAN, A\$1900

CALL -151

1900L

Etc...

In the PRENIBBLE routine they shifted the Primary and Secondary Buffers to BB00. Nothing to worry about.

The WRITE routine does not interest us, we don't want to write to the disk. But it confirms the modification of the Data Field Markers:

B852-A9 DD	LDA	#\$DD	instead of \$D5
B857-A9 96	LDA	#\$96	instead of \$AA
B85C-A9 EF	LDA	#\$EF	instead of \$AD

and the epilogue:

B89D-A9 96	LDA	#\$96	instead of \$DE
B8A2-A9 FE	LDA	#\$FE	instead of \$AA
B8A7-A9 9E	LDA	#\$9E	instead of \$EB

The only other deviation from standard is a DEY for a normal INY.

The WRITE A BYTE subroutine from B8B8-B8C1 is OK. The POSTNIBBLE routine takes in account the shifted buffers.

What we're interested in starts now at B8DC, the READ A DATA FIELD routine. The Markers are changed to:

B8E6-4A	LSR	for EOR \$D5
B8E7-49 6E	EOR	#\$6E for BNE B8DE
B8F0-C9 96	CMP	#\$96 for CMP \$AA
B8FB-C9 EF	CMP	#\$EF for CMP \$AD

They changed the Data Checksum value at B8FF, too and decrement the Y-reg as above.

The epilogue is changed to:

B934-C9 96	CMP	#\$96	for \$DE
B93E-C9 FE	CMP	#\$FE	for \$AA

Normal DOS never checks the 3rd epilogue byte.

The READ an ADDRESS FIELD routine starts at B944 and we find the same modifications as in the Data Field routine. At B954 we have:

B954-4A	LSR	
B955-C9 6E	CMP	#\$6E for CMP \$D5
B95E-C9 96	CMP	#\$96 for CMP \$AA
B969-C9 AB	CMP	#\$AB for CMP \$96

The epilogue is:

B990-C9 96 CMP #\$96 for CMP \$DE

The 2nd byte is not checked with a CMP \$AAA, for we find at
B999-A5 2D LDA 2D
B99B-8D 00 B9 STA B900

It stores the found sector byte inside our code. That's ok, we have seen before with the nibble editor that the Address Markers vary between DC and DD. This is not a problem, they only handle it differently. You will understand.

\$DC shifted right becomes \$6E
11011100 > 01101110 (carry=0)

\$DD shifted right becomes \$6E
11011101 > 01101110 (carry=1)

See what I mean?

The HEAD POSITIONING routine is the normal one, except there are 3 jumps outside of our code at

B9D7-20 9E	A5JSR	A59E
B9E3-20 9E	A5JSR	A59E
B9EA-20 9E	A5	JSR A59E

This code normally JSRs to the MOVE HEAD DELAY routine at BA00. But at BA00 we got 4C 9E A5 JMP A59E. And where we expect the HEAD DELAY routine, there is new code. A quick check at A59E confirms that they relocated the routine.

We have to bring back those bytes inside the available range and it's easy to find some empty space. STOP! Don't cancel any code inside the RWTS, even if it looks like garbage. You never knows, there might be a nibble count somewhere. Think it over, there must be some space, we can use without a problem. Right, there is this WRITE ADDRESS FIELD DURING INIT routine at BC56 to BCC3. You don't want to INIT a disk with their modified DOS? Check above addresses, the complete code is there, they even modified the markers here.

We will install the DELAY routine inside our code:

1D56:A2 11	LDX	#\$11
1D58:CA	DEX	
1D59:D0 FD	BNE	1D58
1D5B:E6 46	INC	46
1D5D:D0 02	BNE	1D61
1D5F:E6 47	INC	47
1D61:38	SEC	
1D62:E9 01	SBC	#\$01
1D64:D0 F0	BNE	1D56
1D66:60	RTS	

And modify the jumps to this code:

1AD7:20 56 BC	JSR	BC56
1AE3:20 56 BC	JSR	BC56
1AEA:20 56 BC	JSR	BC56
1B00:4C 56 BC	JMP	BC56

The ARM MOVE DELAY Tables at BA11-BA28 are OK.

The WRITE TRANSLATE table at BA29 to BA66 shows two changes:

BA29-AA	for 96
BA4F-D5	for DD

The READ TRANSLATE table at BA96-BAFD takes charge of this, we find:

BA96-96	for a normal 00
BAAA-00	for AA
BAD5-26	for D5
BADD-DD	for 26

Now you understand why we weren't able to read a sector, even with correct modified markers in our nibble editor.

We continue to verify the RWTS, buffers and new code at normally unused addresses shouldn't be touched. (see above)

Everything seems ok, till the CHECK for CORRECT SECTOR routine at BE26. Here we find a LDY #\$04 for a normal LDY #\$05 at BE26, and it jumps at BE2B to the new code at BA03.

BE2B-20 03 BA	JSR	BA03	for a normal load of the SECTOR INTERLEAVE Table.
---------------	-----	------	---

A check at BFB8 reveals new code at this location, but no sector interleave table. In the new code at BA03 we find:
BA0C-B9 50 A4 LDA A450,Y

Another access outside of our Bsaved code. Let's look at A450 and surprise, we find not only one, but 2 sector interleave tables.

We have to reinstall them in our code. Be sure to copy both tables; both are accessed by the program. If you copy only one, you will lose some sectors on your copy.

As before, we will use the available space in the Write Address during INIT routine:
1D67:00 0E 0C 0A 08 06 04 02
1D6F:0F 0D 0B 09 07 05 03 01
1D77:08 06 04 02 00 0E 0C 0A
1D7F:07 05 03 01 0F 0D 0B 09
1D87:EA EA EA some NOPs just to be sure

Don't forget to change the load address to the new location.
1B0C:B9 67 BC LDA BC67,Y

The Sector Initialization Map is full of FFs, and even the DOS patch area starting at BFC8 is modified. But don't worry anymore, the work is done. Save the new code
BSAVE RWTS.PACMAN, A\$1900, L\$800

Now relax and let Super IOB do the dirty work. Boot Super IOB and install the SWAP CONTROLLER. Follow the prompts and when you get the JBREAK AT 10010, type:
10010 PRINT CHR\$(4) "BLOOD RWTS.PACMAN ,A\$1900"

Insert the disk you saved your RWTS to, and type:
RUN

Once the modified RWTS loaded, reinsert Super IOB and follow the prompts. DO NOT format, use a blank formatted DOS disk you should always have near you. Last not least, change the boot file name.
RENAME RUNNER,HELLO

And copy a fast DOS, Tony Weishaar's Pron-toDOS will do fine. (Perhaps Beagle Brothers still sell it, if you don't own this excellent utility). Hide your original and enjoy your copy.

Some of you asked for a long softkey with some explanations. Sorry it's only a short and easy one, so complain. Next time I'll try to find

a program with nibble counts, self-modifying code, french dressing a.s.o.

Chaos CT

Advanced Playing Technique for...

Zany Golf

Electronic Arts

For unlimited strokes: first, boot up BASIC.SYSTEM and put a copy of the Zany Golf GS disk in the drive and enter the following: BLOAD/DISK/CODE, A\$2000, L\$13DF, B\$5B81
POKE 8192,234:POKE 8193,234
POKE 8194,234:POKE 8210,234
POKE 8211,234:POKE 8212,234
POKE 9793,234:POKE 9794,234
POKE 9795,234:POKE 13276,234
POKE 13277,234:POKE 13278,234
BSAVE/DISK/CODE, A\$2000, L\$13DF, B\$5B81

Now boot the program and you will have unlimited strokes! To remove the cheat, enter the following from BASIC.SYSTEM: BLOAD/DISK/CODE, A\$2000, L\$13DF, B\$5B81
POKE 8192,157:POKE 8193,78
POKE 8194,66:POKE 8210,157
POKE 8211,78:POKE 8212,66
POKE 9793,157:POKE 9794,78
POKE 9795,66:POKE 13276,157
POKE 13277,78:POKE 13278,66
BSAVE/DISK/CODE, A\$2000, L\$13DF, B\$5B81

The cheat is now removed!

Portions of this file are copyrighted 1989 by Steve Luzynski and Jim Maricondo. Permission is granted for Computist to reproduce this article in any way within Computist magazine.

Steve Luzynski can be reached at:

GENie - SLUZYNski
Compuserve - 76277,2140
AppleLink - SteveL3

Jim Maricondo can be reached at:

GENie - J.Maricondo1
AppleLink - The Mouse

Advanced Playing Technique for...

Neuromancer

Interplay

Cheats, Hints, and Tips

An invaluable player's guide compiled by The Anonymous Guru and Chaos. Thanks to Mr. Xerox, Ice Lord, Chaos, and Relix Distributors. Also, special thanks to Steve Luzynski for many of the tips (NOT the cheats!) included here, and permission to publish them. Without his tips, this article would be mostly cheats. Information on how to use with GS version by Chaos.

Note: differences needed to allow the cheats to be used with the GS version are found in brackets ("[]"). Also, all info for modifying the GS version applies to game 1 only. Use the information on game 1 to modify games 2, 3, and 4 on your own.

Warning: The following contains several very blatant hints and cheats. Do not just read this file if you want the game to be a challenge. Look only at the sections you are having trouble with!

The game Neuromancer can be quite a blast, if you're a weird person who likes demented games that are very out-of-the-ordinary. This is one of those games. It is quite complex, with a number of things you can do, and the way of getting from one place to another in the game gets crazy and hectic after a while. As it says within the game, "It feels like I've been playing this game for 20 years!", and then makes references to being longer than "Wasteland". True, how true.

I think it's quite appropriate that I mention the fact that if you choose to start randomly editing your master disks with the information provided in this article, you're doing so at your own risk, and don't expect anything from us except for sympathy when your disks crash. Be smart - back them up first. (Also, none of the cheats are guaranteed to work for you - I haven't completely tested the IIe ones, just the GS ones which work fine for me.) Also, you will notice that I'm spelling 'softwarez' in the way the game spells it. I am not a surf punk nazi pinko kracker k00l rad awes0me d00dz type person who spells words like that. I would advise obtaining a real copy of the game. In our opinion, we think this game is worth the money to buy it. Software authors have to eat too, ya know.

First off, the listing of items in the game and their hexadecimal equivalents:

Softwarez

- 00 Mimic
- 01 Jammies
- 02 Thunderhead
- 03 Vaccine
- 04 Blammo (virus - do not use)
- 05 DoorStop
- 06 Decoder
- 07 Sequence
- 08 ArmorAll
- 09 KGB
- 0A ComLink
- 0B BlowTorch

- 0C Probe
- 0D Drill
- 0E Hammer
- 0F Python
- 10 Aci
- 11 Injector
- 12 DepthCharg
- 13 Concrete
- 14 EasyRider
- 15 LogicBomb
- 16 Cyberspace
- 17 Slow
- 18 BattleChess
- 19 BattleChess (virus - do not use)
- 1A Scout
- 1B Hemlock
- 1C Kuang Eleve

CyberDecks

- 1D Hiki Gaeru
- 1E Gaijin
- 1F Bushido
- 20 Edokko
- 21 Katana
- 22 Tofu
- 23 Shogun
- 24 188BJB
- 25 350SL
- 28 UXB
- 2A ZXB
- 2B Cyberspace II
- 2C Cyberspace III
- 2E Cyberspace VII
- 2F Ninja 2000
- 30 Ninja 3000
- 31 Ninja 4000
- 32 Ninja 5000
- 33 Blue Light
- 34 Samurai Seve

Skill Chip

- 43 Bargaining
- 44 CopTalk
- 45 Software Analysis
- 46 Debug
- 47 Hardware Repair
- 48 ICE Breaking
- 49 Evasion
- 4A Cryptology
- 4B Japanese (I don't know what this skill is for.)
- 4C Logic
- 4D Psychoanalysis
- 4E Phenomenology
- 4F Philosophy
- 50 Sophistry
- 51 Zen
- 52 Musicianshi

Special Items

- 53 CyberEyes!
- 56 Matrix Restaurant Guest Pas
- 59 Joystick
- 5E Caviar
- 5F Pawn Ticket
- 60 Sense/Net Security Pass
- 61 Zion Ticket
- 62 Freeside Ticket
- 64 Chiba City Ticke
- 65 Gas Mask
- 67 Sak

Ok, how to use these. Within the file DATA1 on the NEUROMANCER1 disk are the 5 'saved games', all of which are about 2 blocks each. One original 'virgin' game is kept so the game can create saved games from the start. Your 4 saved games begin immediately after that initial one. For our purposes, you will have to, at least, go into the game, and as soon as you get there, save the game to a location. Game 1 starts at Block \$002D (45) roughly [or \$00FD on the GS version of the soft]. You will know when you are there, because you will see your character's name somewhere around bytes \$005AB0 or so (or byte \$B0 within the block) [starting at around \$1B5 on the GS version]. Just before your characters name you will see a bitmap, of sorts, that looks something like this:

```
00 00 00 00 00 00 00 00
5F 00 00 00 FF 00 00 00
FF 00 00 00 FF 00 00 00
FF 00 00 00 FF 00 00 00
FF 00 00 00 FF 00 00 00
FF 00 00 00 FF 00 00 00
FF 00 00 00 FF 00 00 00
FF 00 00 00 FF 00 00 00
FF 00 00 00 FF 00 00 00
FF 00 00 00 FF 00 00 00
FF 00 00 00 FF 00 00 00 etc...
```

This is an inventory of what your character is carrying. The chunk of bytes above happens to be about 1/2 of a fresh virgin character... you can see (if you check the inventory chart above) that all the character is carrying at the very start of the game is the pawn ticket (hex \$5F). In order to add items into the inventory, change the dead bytes (hex \$FF) to any one of the bytes in the tables listed above. If you wanted a security pass, stick a \$60 in one of the slots in the above array. [GS users: Inventory for game 1 is found on block \$FD, bytes \$125 thru \$19D (or so I can tell - it may look like thru \$1A1, but I wouldn't risk it)]

The same format is used for softwarez that are in your deck. This array is located AFTER your characters name, starting at around \$5B00 in block \$2D [block \$FE, for game 1, bytes \$024 thru \$07D for GS version]. The software arrays are in blocks of three bytes... referring to the above array...

```
FF 00 00 FF 00 00 FF
00 00 FF 00 00 FF 00
00 FF 00 00 FF 00 00
FF 00 00 FF 00 00 FF
00 00 FF 00 00 FF 00
00 FF 00 00 FF 00 00
FF 00 00 FF 00 00 FF
00 00 FF 00 00 FF 00
00 FF 00 00 etc...
```

[GS note: 'Softwarez' are stored slightly different in the GS version - a deck with no softwarez looks like (on byte \$FE, now):

These have no use!

```
  \  \  \  \
020 - FF 01 00 00 FF 00 00 00 FF
      00 00 00 FF 00 00 00
030 - FF 00 00 00 FF 00 00 00 FF
      00 00 00 FF 00 00 0
```

Fill in the FF's (except the byte \$020 one) with the software's hex ID, the 00 (after the FF) with the software's version number, and the second 00 with the number of errors in the software. I've got no idea what the third 00's for!

The software title (Comlink, Thunderhead, Logicbomb) is the byte that is now a dead byte (\$FF). The first byte following that is the version number of the software (\$06 for 6.0, \$11 for 17.0). The next byte is the amount of errors in that software. People who have been into cyberspace and confronted an AI that kicked the sh-t out of them, know that your software gets destroyed and you need a real high level of Debug to fix it. This byte, when non-zero, means the software is trashed and requires debugging. The higher the value of that byte (\$01 - \$FF), the more damage there is to it. Suppose you get whipped in cyberspace, reset these bytes to \$00 and you're software is good as new. So therefore, an example of a deck full of software:

```
00 00 00 0A 06 00
01 02 00 08 11 0A
0B 01 00 FF 00 00
FF 00 00 FF 00 00
FF 00 00 FF 00 00
FF 00 00 FF 00 00
FF 00 00 FF 00 00
FF 00 00 FF 00 00
FF 00 00 FF 00 00 etc...
```

If that was your array, you'd have the following in your deck:

Comlink 6.0, working (\$0A, \$06, \$00)
Jammies 2.0, working (\$01, \$02, \$00)
ArmorAll 17.0, non-working (\$08, \$11, \$0A)
BlowTorch 1.0, working (\$0B, \$01, \$00)

Using these arrays can be helpful if you happen to have a piece-of-crap deck with only 5 or 10 RAM... you can fill the thing up to the brim, regardless of which deck you have. But why not get for the best deck in the first place?

In order to upgrade your skills, you'll have to give yourself the skill chips by adding them into your inventory, go into the game and Operate the chips (install them), save the game (make sure it's the same game that you're editing) and go back into your editor... when you're there, check for the following bytes (they are in between the inventory array and the software array)... But before that...

```
FF FF FF FF FF FF FF
54 59 20 52 45 58 00 36
37 38 39 D0 07 00 D0 07
04 0C 28 00 00 00 00 00
00 00 A8 07 00 00 00 00
00 00 00 00 00 00 00 00
01 04 40 D0 07 00 40 38
00 00 40 4B 00 00 C0 E8
03 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 etc...
```

[GS notes: all for saved game one, of course!] Skill levels found on block \$FD, bytes \$1A5 thru \$1B4, for each of the 16 possible skill chip 'spots' (or chips in the spots) you can have.

Health found on block \$FD, bytes \$1C3 thru \$1C4 in reverse hex (2000 = D0 07 in reverse hex).

Sold body part record found on block \$FD, and starts at byte \$1C8. Don't ask me how far it goes - I've never had to sell any body parts for quick ca\$h! Also, this byte is approximate - I'm pretty sure but not 100% sure.

Money on hand found on block \$FD, bytes \$1CF thru \$1D1 in reverse hex (e.g. 98 96 7F = 9,999,999 (in regular hex), so in order to have \$9,999,999 on hand enter \$7F in byte \$1CF, \$96 in byte \$1D0, and \$98 in byte \$1D1, to have the most money possible on hand at one time!).

Money in PAX banking account found on block \$FD, bytes \$1D2 thru \$1D4 in reverse hex (see above for explanation).

The first line of bytes contains your characters name, so if you can see that you're in the right place, as the bytes we're after are the lines immediately after it. Notice the 'D0 07' sequence. This is your characters constitution (health). They are in normal hi-low format, so your characters health would be \$07D0, or 2000. As with games with a 'current/maximum' type thing, as does this game. The first two (\$D0 and \$07) are followed by a zero, and then repeated... this is the current/maximum deal. Immediately after is a three-byte sequence (\$04 \$0C \$28) which holds your current location. Just past those

three (which are all \$00 in the example) is an array of the body parts you have sold. If any of those 3 or 4 bytes are non-zero, change them to \$00 and your body parts will be returned to you. If you didn't know that you can sell body parts to raise cash, you should keep playing the game some more... Now here's a good one for you... On the following line are the bytes \$A8 and \$07. These are your cash on-hand in hi-low format. Right after that are a hi-low sequence of the cash you have in the PAX bank account.

Anyway, back to the skill upgrades. The line immediately before your characters name contains your skill levels, per chip. So...

```
FF 00 00 00 2E 19 00 00
FF 00 00 00 FF 00 00 00
00 00 00 00 00 00 00 00
11 11 11 11 11 11 11 11
11 11 11 11 11 11 11 11
54 59 20 52 45 58 00 36
37 38 39 D0 07 00 D0 07
04 0C 28 00 00 00 00 00
00 00 A8 07 00 00 00 00
00 00 00 00 00 00 00 00
01 04 40 D0 07 00 40 38
00 00 40 4B 00 00 C0 E8
03 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 etc...
```

The first line in this example is the last line of your inventory array. The next line contains \$11 bytes, each of which is a skill level of a certain chip. So therefore, starting at the first \$11, you can move along and give yourself whatever skill level you want. After messing with these values a bit, we've come to the conclusion that the 'highest' value the game will recognize is \$0C, or 12. This is because every time you break an AI, certain skill chip levels will increase. If you have 65 for a skill level, the game will try and upgrade it and crash, just for you.

Hints and Tips

Part 1: Starting out.

When you start out, you are in a seedy little bar, having just woke up from a hard night's sleep in a plate of spaghetti. You'll need to pay the bartender for your food, and question him as much as you want. You cannot get back into the bar once you leave so make sure you finish all your business. Read all the articles on the PAX news system and the Bulletin board.

Your next priority is to get back your deck. Go to the pawn shop and ask for your deck back, but say you don't have the money. He'll give you the deck for free. Now, save your game and walk around. Enter every building you can and ask the people there about anything you can think of.

Now you should go to the Gentleman Loser. Talk to the woman. (If you read the PAX messages like you were supposed to, you'll know what to ask about.) You should try to get some passwords and link codes from her.

Now, play with your deck. Use the link codes and passwords to get into various databases. (From here out, databases will be referred to as DB's.) Make sure to explore everything very heavily. Whenever you find out something you might consider useful, write it down! Also save your game before and after every major decision. You should be finding out some very useful info as to what exactly you should be doing.

From here on out, it's up to you.

Part 2: ROM Constructs, ca\$h, and a few questions...

Ok, having gone ape on this game, I can offer some interesting tips and clues and things like that. One of the best things you should get is a RomConstruct. These are little suckers that basically emulate yourself, and is kind of an alter-ego. There are three different ones, and all do the same thing, but they act a little differently. Dixie Flatline is the most reliable. The other two are Toshiro Mifune and Rombo. Rombo acts like a Stallone rip-off, and Toshiro is Japanese. No big deal. They're all the same, and they do the same things, they only express themselves differently. To get these, follow these steps:

Get into the Hosaka link (ComLink 5.0 minimum is required for this) with a second level password (FUNGEKI). List out the employees and edit one of them to contain your name and BAMA ID (which is always 056306118). This will allow you to get past the security guard in the hi-tech sector. (Tell him you work for Hosaka and you're in.) Next, get into the Tactical Police link with the second level password (SUPER TAC). Edit one of the arrest warrants to include the Microsofts owner's name, Larry Moe. His BAMA id can be found in the Fuji link under the employee listings (hint: 062788138). (Note: before you get rid of him, buy the CopTalk skill if you want, cause he sells it! Or, of course, you could just edit the skill into your inventory!) This allows you to get into the back room of his shop. In the back room of Microsofts, you will meet Lupus, the leader of the Panther Moderns. Ask him about Sense/Net and he'll sell you a security pass to get into Sense/Net so you can pick up your RomConstruct. (Or, you could edit the pass in.) Go to the Sense/Net building, give the computer your pass, and pick up the construct!

The RomConstruct ID's that the Sense/Net computer will ask you for are as follows: Dixie Flatline is 0467839, Toshiro Mifune is 6905984,

and Rombo is 5521426. I think these can all be found in the Sense/Net Library DB.

Other problem encountered is getting credits for cyberspace. Of the many ways to get money, here are a few:

Upload Comlink 6.0 to the Hosaka Software library. 7500 credits.

Check the PAX for the Ad from Armitage. Answer it. You will be credited with a goodly sum of money. You will also get arrested if you go near a Lawbot. It's a \$500 fine, but after that you're free.

Illegal bank transfers. First, go to your account on the PAX. Download about \$5,000 to your chip. Enter the Bank of Zurich DB (either through Cyberspace or the conventional way.) Create an account, and write down the number! Then, you have a lot of different ways to go. You can ask Lupus about Banking, or just keep reading.

OK, your first number is for Bank Gemeinschaft. Enter it's DB and go to Account operations. (You need the highest level of access for this sort of thing. Use cyberspace, or the second level password.) The number you want to transfer from is 646328356481. The link code for the destination bank is BOZOBANK. The next number is for the Bank of Zurich. Yep, that's the bank you're keeping your own money in. It's ok, you CAN'T get caught. That number is 712345450134. Your last number is for the Bank of Berne. The Authorization number is LYMA1211MARZ. The Account number is 121519831200. In all these transfers, pick up ALL the money! It's yours to take! All in all, you'll end up with around \$550,000!! Keep all but a couple thousand in your PAX account. (It's the most convenient.) Carry around \$1200 with you at all times, and keep a little bit in the Bank of Zurich, for a rainy day!

Walk over to the Hi-Tech Zone after 'signing' up with Hosaka. When the computer asks what company you're employed by, tell Hosaka. It will let you past. Go into the building. You will receive a message that you have gotten your paycheck. It will be around \$15,000. You can get your check twice. (You have to wait a week in between.)

You can always sell your body parts at Mod Bods... but this is not a good idea at all.

Get a hold of the BattleChess software and beat the WORLDCHESS link at its own game. You'll get a bunch of credits the first time and a whole lot more the second time. Warning: you'll lose the third time until you find a higher version of BattleChess. Then again, you can always load up Prosel's Block Warden (or Copy II+) ... hint hint.

Q: What should I ask people about?

A: Ask about skills, upgrades, Matrix, Cyberspace, Neuro-mancer, hardware, banks, softs, and anything else you can think of. Try the words off of the PAX wheel.

Q: What is there to do in Maas Biolabs?

A: First, you need to break into the Maas Biolabs DB in zone 6. Read all the messages, and shut off all security systems and unlock the door. Go to Julius Deane's office. Ask about Hardware. Buy a gas mask. Use it. Go across the street to the labs. Answer in an encouraging manner to the computer. Get the CyberEyes system implanted in your own skull. You don't really need to do this, but it's kinda neat. You can really load up on viruses and other softs!

Part 3: more assorted tidbits that you'd probably find out anyway after playing for a while...

BattleChess 4.0 and Blammo 1.0 are virus programs. Don't use them.

AI's are present in a lot of bases in cyberspace. You need to get your skills up since skills are the only thing that kill AI's. This, of course, is open to exceptions... The Musabori base contains the AI Greystoke which can only be killed by the 'Hemlock' software. The last AI to kill, the Neuromancer, which guards Allard Tech, must be killed with the Kuang Eleven software AND certain skills.

The end-game sequence is amusing. When you think you've won, you really haven't. When Ice Lord and I were on the phone (I was calling to brag about finishing this game) I realized I hadn't, and spent the next hour fighting with the Neuromancer (the jerk).

Part 4: Database link codes and passwords.

Note: The password is for the level of access shown by the Level number. If the Level one password is not given, it is on the title screen for that DB. However, you should always use the highest access password you have access to. Also, some higher level passwords for some bases are not given. That is because by the time you need that level of access, you can get in through Cyberspace. An \$\$ indicates there is a fee for access. -no password means to use Sequencer 1.0 to access this base.

Database Name	Link Code	Password(s)	Level
Panther Moderns	CHAOS	Mainline	1
Cheap Hotel	CHEAPO	Cockroach	2
Gentleman Loser	LOSER	Loser	2
Regular Fellows	REGFELLOW	Visitor	1
Consumer Review	CONSUMEREV	(see note)\$\$	1
Asano Computing	ASANOCOMP	Vendors	2
World Chess Fed.	WORLDCHESS	Member	2
Psychologist	PSYCHO	Babylon	2
FreeMatrix	FREEMATRIX	Cim	1
InternalRevService	IRS	Audit	2
Hosaka	HOSAKACORP	Fungeki	2
Fuji Electric	FUJI	Uchikatsu	2
Musabori	MUSABORIND	Subaru	2
Hitachi Biotech	HITACHIBIO	Biotech	2
Software Enf.Agency	SOFTEN	Permafrost	2
Chiba Police	KEISATSU	Supertac	2
Bank of Zurich	BOZOBANK	-no password	1
Tozoku	YAKUZA	Yak	1
NASA	VOYAGER	Apollo	1
East.Sea.Fission	EASTSEABOD	Longisland	1
Bank Gemeinschaft	BANKGEMEIN	Verboten	2
Copenhagen U.	BRAINSTORM	Perilous	1
Justice Dept.	JUSTICE	-no password	

Part 5: Cyberspace

In Cyberspace, you should be very sure to finish all the DB's in a zone (Except for the one's with AI's in them in Zones 0 & 2.) before moving on to the next one. Check the Cyberspace table below to make sure. Save your game before entering each DB and save again after if you make it. Go back to the real world to recover if the EEG ever turned yellow or red while you were trying to break the ICE. Make sure you use your ICE Breaking skill before you do anything or you'll get creamed. Look at the ICE Breaking Section for the best way to go about ICE Breaking.

Note: In the following section, the Zone(ZN) is the leftmost number on your cyberspace deck's co-ordinate display, then comes X, and Y is the next one over. If there is an AI, you should not attempt to enter these bases until you have the necessary skills.

Database Name	Z	X	Y	AI Name	AI STR	ICE STR
Asano Computing	0	16	112	None	—	
Consumer Review	0	32	64	None	—	
Psychologist	0	96	32	(unknown)		96
Cheap Hotel	0	112	112	None	—	
Regular Fellows	0	208	32	None	—	
Panther Moderns	0	224	112	None	—	
World Chess Fed.	0	160	80	(unknown)		84
FreeMatrix	1	352	112	(unknown)		150
IRS	1	272	64	None	—	
Software Enf.Agency	1	352	64	None	—	
Chiba Police	1	288	112	None	—	
Tozoku	1	480	80	None	—	
NASA	1	448	32	(unknown)		132
Gentleman Loser	1	416	64	None	—	
Copenhagen U.	1	320	32	None	—	
Justice Dept.	1	416	112	None	—	
Hosaka	2	114	160	None	—	260
Fuji Electric	2	112	240	None	—	260
Musabori	2	208	208	Greystoke	25000+	260
Hitachi Biotech	2	32	192	None	—	260
Bank of Berne	3	336	160	(unknown)		400
Free Sex Union	3	288	208	(unknown)		400
Turing Registry	3	432	240	None	—	400
Screaming Fist	3	464	160	None	—	400
DARPA	3	336	240	None	—	400
Gridpoint	4	160	320	None	—	800
SENSE/NET Library	4	48	320	None	—	800
Bell Europa	5	384	288	None	—	320
Bank Gemeinschaft	5	304	320	None	—	
Bank of Zurich	5	336	368	None	—	
Nihilist	5	416	368	None	—	
INSA	5	448	320	None	—	
Maas Biolabs	6	112	480	Sangfroid	20000+	1100
KGB	6	112	416	Lucifer	20000+	1100
Phantom	7	320	464	Phantom	24500+	2000
Tessier-Ashpool	7	384	416	Wintermute	23000+	2000
Allard Technology	7	432	464	Neuromancer	26000+	2000

Part 6: Notes on DB's.

Database Name	Comments
Panther Moderns	This is a good place to get information and software (but you can only get warez for the better decks.
Cheap Hotel	Here, you can erase your bill so you can get in the building. You can also order room service.
Regular Fellows	Info and software. Definitely access this early on.
Consumer Review	Reviews all the available decks. Will charge you \$300, so: Save game, access it, write down the info, restore the game. You get the info for free that way.
Asano Computing	Lists the link codes for some companies that are already in the above table. Read the messages anyway.
World Chess Fed.	A good way to earn some easy cash. Access the Regular Fellows DB first to get a chess program.
Psychologist	Has some interesting messages. DO NOT post anything!
FreeMatrix	Has a very dangerous program. Do not download anything from it.
InternalRevService	Good messages and some other functions..
Hosaka	Very good place to get software. Also read the messages.
Fuji Electric	Read the messages.
Musabori	Read the messages.
Hitachi Biotech	Read the messages.
Software Enf.Agency	Good source of ICEBreakers and messages.
Chiba Police	Get people out of your way by having them arrested.
Bank of Zurich	It's a Bank. Read the messages and see the Money section below.
Tozoku	Read all the messages.
NASA	Read the messages.
East.Sea.Fission	Read the messages. Good software.
Bank Gemeinschaft	It's a Bank. Read the messages and see the Money section below.
Copenhagen U.	Read the messages.
Justice Dept.	Read the messages.
Gentleman Loser	Good info and great softs to get you started off.
Bank of Berne	It's a Bank. Read the messages and see the Money section below.
Free Sex Union	Read the messages
Turing Registry	Read the messages. Also has skill chip upgrades you: MUST have to be able to kill AI's.
Screaming Fist	Info and great software.
DARPA	Info and great software.
Gridpoint	Info and great software.
SENSE/NET Library	Part numbers for ROM constructs. It's still up to you to get one, though.
Bell Europa	Info and Software. The easiest one to get into in Zone 5, so do it first.
Nihilist	Great software. Ignore the messages.
INSA	Info and awesome software.
Maas Biolabs	Access the security system for the building in South Chiba.
KGB	Lots of great software and good info.
Phantom	Valuable experience in ICE Breaking and AI killing. Otherwise, a dull one.
Tessier-Ashpool	See above. Same situation.
Allard Technology	This should be the last thing you do. If you kill Neuromancer, you win.

Part 7: ICE Breaking: How to do it.

Softs	Location	Function
Comlink 1.0	your deck	Basic communications package.
Comlink 2.0	Edo's	Upgrade to above.
Comlink 3.0	you'll find it	With each upgrade, you can access
Comlink 4.0	you'll find it	more of the DB's. They are easy to
Comlink 5.0	EASTSEABOD	find if you look around.
Comlink 6.0	TOZUKU	This is the last version.
Decoder X.0	Various	This is a first level ICE Breaker.

Blowtorch X.0	Various	This is a first level ICE Breaker.
Hammer 1.0	Various	This is a first level ICE Breaker.
Hammer 5.0	Gridpoint	This is a first level ICE Breaker.
Hammer 6.0	INSA	The best version of this program.
Drill X.0	Various	A capable, but weak 2nd level soft.
Doorstop 1.0	Various	A good 2nd level soft.
Doorstop 4.0	INSA	The best version of this program.
Concrete X.0	Various	A very good 3rd level program.
Concrete 5.0	KGB	The best version of Concrete there is
DepthCharge X.0	Various	The 2nd best 3rd level soft.
DepthCharge 8.0	KGB	The top version of this soft. Great!
LogicBomb 3.0	Nihilist	The best 3rd level there is.
LogicBomb 6.0	KGB	The very best ICE Breaker there is!
Slow 4.0,5.0	Nihilist,KGB	They slow down the ICE so it can only shoot a few times before you kill it
Jammies X.0	Various	See above. Not as good as Slow.
Probe X.0	Various	Tells you the characteristics of a DB. How good it is depends on the #.
Acid X.0	Various	A virus. Helps you attack a DB.
Acid 5.0	Bell Europa	The best version of Acid there is.
Thunderhead 2.0	SEA	Another virus.
Thunderhead 4.0	Bell Europa	The best version of Thunderhead.
Python 5.0	Nihilist	A virus. Lower versions are junk, but this one is great.
Injector 5.0	INSA,KGB	Another good virus.
ArmorAll X.0	Various	Restores your deck's shielding.
ArmorAll 3.0	INSA	Allows 3 uses.
ArmorAll 4.0	KGB	Allows 4 uses.
EasyRider 1.0	Various	Lets you skip over Zone boundaries. A must have!
CyberSpace 1.0	Various	A replacement for Comlink 6.0 when you want to enter cyberspace.
Hemlock		Use this to 'kill' Greystroke AI

Any other softs you come across should be regarded as dangerous and should not be used unless you are very careful. KGB, for example, will take you right to the KGB DB in level 6. Unless you are ready for it, you'll get creamed.

The process

First, before you even enter a base, save the game! You should use one save spot for all your saves in Cyberspace. (I used number 4.) That way, if you're getting creamed, you can restore the game to Cyberspace and not have to re-log on.

As soon as you enter, use your ICE Breaking skill. Then, shoot the ICE with either Slow or Jammies. This will make the ICE respond very slowly. With a high version of Slow or Jammies, you can hit the ICE 5 times before it hits you even once. Next, shoot it with the highest numbered virus you have. This will make things go much quicker. With a really good virus from zone 6 or 7, you can break into Zone 0 DB's without ever firing a shot!

Then, start hitting it with all the ICE Breakers you have. Don't use each one more than twice in a row, though. They lose their effectiveness. Use the very highest, best versions of each breaker. Watch the ICE bar over on the very right side to see which ones hurt it and which ones don't. When the bar is all white, the next shot or hit from the virus will kill it. When it dies, watch to see which disk gets accessed. If it goes to side 3 or 4, you did it! If not, you have broken into a base with an AI in it. Then, you will have to a) use your Evasion skill to get away, or b) Kill the AI.

Part 8: AI's

If you are having MAJOR problems getting past a certain AI, fear not. You can get rid of him with little difficulty and some help from your friendly sector editor (but of course!). The following is a listing of bases found in cyberspace that have AI's associated with them, and what is required to kill them off. These weaknesses could also have been found out by you, using Psychoanalysis skill on each individual AI.

Database	AI Name	Main Weakness
World Chess	Morphy	\$09 - Logic
Psychologist	Chrome	\$0C - Philosophy
Free Matrix	Sapphire	\$0D - Sophistry
Musabori	Greystoke	\$1B - Hemlock Software
N.A.S.A.	Hal	\$09 - Logic
Free Sex Union	Xaviera	\$0B - Phenomenology
Bank of Berne	Gold	\$0C - Philosophy
Maas Biolabs	Sangfroid	\$0B - Phenomenology
KGB	Lucifer	\$09 - Logic
Phantom	Phantom	\$09 - Logic
Tessier-Ashpool	Wintermute	\$0D - Sophistry
Allard Tech	Neuromancer	\$1C - Kuang Eleven

Regular Fellows	none
Consumer Review	none
Asano Computing	none
Cheap Hotel	none
Panther Moderns	none
I.R.S.	none
Fuji Electric	none
Tacticalice	none
Hitachi Biotech	none
Copenhagen University	none
S.E.A.	none
Eastern Seaboard	none
Gentleman Loser	none
Tozoku	none
Hosaka	none
Bank Gemeinschaft	none
Bank of Zurich	none
Central Justice	none
Bell Europa	none
Nihilist	none
I.N.S.A.	none
Sense/Net	none
Gridpoint	none
D.A.R.P.O.	none
Turing Registry	none
Screaming Fist	none

[GS note: I have not actually tested 'editing' out AIs, (haven't gotten that desperate, yet) but if you are interested, the AI 'none's or records begin on block \$11E on the GS version.]

OK, now say that you're bound for destruction in front of one of these AI's...no problem. Put your cheating cap on and grab your sector editor... read in block \$49... the tables of the AIs and databases start at around byte \$92BB, and are in this format:

```
00 48 00 01 FF FF FF 00 .H.....
00 6E 6F 6E 65 20 20 20 .none
20 20 20 20 20 00 43 6F 6E .Con
73 75 6D 65 72 20 52 65 sumer Re
```



```
76 69 65 77 20 00 00 00 view ...
00 C8 00 00 00 48 00 01 .H...H..
FF FF FF
```

The above just happens to be Consumer Review, and there is no AI present (see the 'none' in there?)... Now, a database guarded by an AI looks like:

```
00 B8 0B 03 07 .8...
D0 07 FF 00 C0 0B 00 00 P...@...
.4E 65 75 72 6F 6D 61 6E Neuroman
63 65 72 00 41 6C 6C 61 cer.Alla
72 64 20 54 65 63 68 2E rd Tech.
20 20 20 20 00 AB 02 00 .+...
00 00 00 00 00 00 00 00 .....
```

As you can see here, the base is Allard Tech, which is protected by the Neuromancer. Before the 'Neuromancer' bytes (\$4E 65 75 72...) you can see all sorts of interesting goodies. Start changing the bytes from where you see the \$C0 and \$0B. Change it to look like the following:

```
00 B0 04 02 07 .0...
90 01 FF FF FF FF 00 00 .....
6E 6F 6E 65 20 20 20 20 none
20 20 20 00 41 6C 6C 61 .Alla
72 64 20 54 65 63 68 2E rd Tech.
20 20 20 20 00 AB 02 00 .+...
00 00 00 00 00 00 00 00 .....
```

Do this, and your all set. What we did is simple: we swapped out the initial record bytes which lowered the ICE strength and trashed the AI before you even got to it. Even though we used the Neuromancer AI as an example, it is the only AI that CAN'T be edited out of the game, simply because the code around it is all part of the end-game sequence, and altering the Neuromancer causes the end-game to be destroyed (not in terms of the game, but physically in the memory of your computer). So, simply, count back 12 bytes from the beginning of the AI name and poke in: B0 04 02 07 90 01 FF FF FF FF 00 and replace the AI name with 'none' (lower case is important!) and fill the end of field with \$20 (spaces). Go in, melt away the ice, and no more AI. This is helpful with high-level AIs... not that cheating is the best way to do it, but if you're looking to get through the game, its the easiest.

For those of us who don't want to cheat on AI's:

Before you even try to kill an AI, you must have: Zen, Sophistry, Psychoanalysis, Phemonology, Logic, and Evasion. Use Evasion if you're getting beat to death.

Zen: Calms your mind down. If the EEG turns yellow or red, use this skill to get back to blue.

Psychoanalysis: Tells you the AI's weakness. It's weakness, when you use it, will make it say something the first time, and after that, hurt it badly.

Use the weakness skill until the computer beeps at you when you try to use it. Then, use each skill twice on it and then cycle through them again. If you get beeps on every skill and the AI still isn't dead, split fast!

As soon as you can, go to the Turing Registry DB and get all your skills upgraded as high as you can. Further upgrades come whenever you successfully use a skill.

Part 9: Skills: What they are

Skill Name	Where to get	Use
Coptalk	The Matrix	Use on the cop in Donut World.
Bargaining	Various	Never needed it myself.
Cryptology	Shiba	MUST HAVE! For decoding passwords.
Musicianship	Julius Deane	Only use is on Zion Cluster.
Software Analysis	The Matrix	Tells you what your softs do.
Debug	The Matrix	MUST HAVE! Lets you fix bugs.
Hardware Repair	Shiba	Lets you fix your UXB at the start of the game.
ICE Breaking	Various	MUST HAVE! You can't win without it.
Evasion	Various	MUST HAVE! See AI Section above.
Zen	House of Pong	MUST HAVE! See AI Section above.
Sophistry	House of Pong	MUST HAVE! See AI Section above.
Logic	Julius Deane	MUST HAVE! See AI Section above.
Psychoanalysis	Julius Deane	MUST HAVE! See AI Section above.
Phemonology	Various	MUST HAVE! See AI Section above.

You may run across other skills in the course of the game. Try them and see what they do. The above is all I ever needed so you can win without any others you may find.

Roger Williams OR

I have finally gathered my notes together on several educational programs I had the pleasure of cracking for several of the local school teachers this summer.

Notes on programs published by Micrograms

I made backups for several elementary programs published by this company. When I first looked at the program I thought I was going to be in for a really tough battle. The disk is prevented from being copied by COPYA or many other 'normal' whole disk copiers by having track 1 or 2 being totally unformatted. There are no files on the disk (that I could find) and the RWTS (if there is one??) was not in the normal location. Yuchh! To start out, I did the obvious thing and made a 'backup' with Copy II+ and booted to see how far it got before it discovered the change in the disk. It never did. The programs ran perfectly. It seems that the publisher figured that the only copy programs that 'computer-ignorant' educators would have would be 'official' Apple copy programs. The whole point of this is to point out that all is not always what it appears to be in the world of protection. Always try the easy way first.

Softkey for...

Aesop's Fables IIe

Unicorn

When I was given this program, I was rather surprised that it actually existed. I had never seen any advertisements for any IIe Unicorn programs. Its only protection is altered epilogs. To avoid the issue of the program checking for the original epilogs, I substituted normal DOS and all worked fine.

1. Load COPYA from System Master.
CALL -151 to enter the monitor
B942: 18 tell DOS to ignore epilogs
3DOG return to BASIC with DOS still active
RUN start COPYA

2. Copy a normal DOS to Fables copy. Use a fast DOS or Master Create from the system disk.

Note: Most Sofkeys using COPYA tell you to delete line 70. In most cases this is not necessary as the program that loads in from line 70 does not affect the DOS area. The only times that deleting

line 70 is necessary is if you SUBSTITUTE a new value to be compared in the read portion of DOS AND modify the COPYA code to not read all of the tracks or ignore 'unreadable' tracks. Most softkeys use an ignore approach to unusual situations rather than actual substitutions. The reason this is usually true is that most COPYA softkeys copy the entire disk and to substitute values would make at least track 0, sector 0 unreadable.

Softkey for...

Kittens, Kids, and a Frog Scuffy and Friends

Hartley Software

These programs also have altered epilogs of DA AA. (Normal epilogs are DE AA) However it was not possible to substitute another DOS so the abnormal DOS had to be modified to work with normal epilogs.

1. Load COPYA from System Master.
CALL -151 to enter the monitor
B942: 18 tell DOS to ignore epilogs
3DOG return to BASIC with DOS still active
RUN start COPYA

2. Get out your favorite sector editor and make the following edits.

Trk	Sci	Byte	From	To
0	2	9E	DA	DE
0	3	35	DA	DE
		91	DA	DE

Softkey for...

World History Adventure

Intellectual Software

The original that I got on this one had already suffered some abuse and a couple of the tracks were bad. (Some of the files had bad checksums and unreadable data. Fortunately they are text files and it is hoped that they will be fixable now that the disk is unlocked.)

Because of the bad files, I had to use Demuffin Plus to pull off the files, but I think that Super IOB with a swap controller would have worked just fine if the files had not been damaged.

Tracks 3-22 had changed prologues. The normal prologues are D5 AA 96 for the address

header and D5 AA AD for the data. The changed headers were BA AA 96 and D5 AA 96. This particular arrangement causes all sorts of problems with bit copiers as well as normal copiers because the copier finds the D5 AA 96 header on the data and assumes that it has the sector address header.

To crack this one, I booted a normal disk, initialized a new disk with HELLO as the boot program on both sides, booted the original, slipped into the monitor, and moved the RWTS to a safe place, booted my 'copy disk', BLOADED Demuffin Plus, moved RWTS back to its normal position and copied the files to the new disk. I had to make some additional changes to DOS in order to get all the files off the original disk because of the damage, but that has nothing to do with the normal softkey. I will however mention the other changes in a note at the end in case some of you someday have a similar situation. As I said before I believe that the swap controller with Super IOB would have worked on an undamaged copy of the same program.

Since several other authors have in recent issues of Computist explained the use of both the Swap Controller and Demuffin Plus in some detail; I will skip the Cookbook methods and give some information which may be new to some of you.

Determining when individual files can be removed from a Protected Disk

The following information will hopefully help some of you readers understand how we all can determine when individual files can be removed from disks as opposed to whole disk copying. Two reasons that you might not wish to copy the entire disk would be if the disk were not entirely formatted or if the disk had suffered some actual damage. I am sure that there are other occasions when file copying is easier/more rewarding than attempting to copy the entire disk.

For those of you not familiar with DOS 3.3 (and anyone else who has not tried this before), I offer the following experiment. Boot a DOS 3.3 disk (the system master is best if you have one), break into basic if the disk doesn't put you there, and enter the monitor with 'Call -151'.

At this point we pause for a little background information. DOS is merely a collection of small programs or routines that are connected to BASIC with command words and several look-up tables. When you type CATALOG from basic, DOS grabs the command to see if it's a DOS command or merely a BASIC command. If it's a DOS command, the DOS looks up the address of the program that the command activates, and runs it. We now return you to our experiment.

Generally, in Computist, most of the discussion of DOS circulates around the addresses between B800 and BFFF in memory. We are going to be looking at some other memory.

The memory location A56E is the beginning of the catalog program. At this point, with your disk still in the drive, type A56EG (which means A56E GO) and the drive will start up and the catalog will appear and you will be returned to the monitor when the listing is finished.

The actual CATALOG program is fairly short because it runs another program. My purpose here is not to dissect DOS (Beneath Apple DOS does that much better than I ever could) but rather to help you see another tool in the deprotection game. With that in mind I give you only the listing of the Catalog program so that you will be able to recognize if it exists on the program you are attempting to crack.

Type the following:

A56EL

And you will see the following (L)isting:

```
A56E: A9 06 LDA #06
A570: 20 AA A2 JSR A2AA the other 'program' I
                          mentioned
A573: AD BF B5 LDA B5BF
A576: 8D 66 AA STA AA66
A579: 60 RTS Return from whence you
                          came
```

Several older protections changed the command words but not the actual routines. Example: If I wrote a 'DOS program' and changed the CATALOG command to KITALOG or GOLATAC and you type CATALOG, you will at best get a syntax error message and at worst, nothing will happen. But if you enter the monitor and see that the memory area between A56E & A579 looks like this; you will be free to catalog the disk by typing A56EG from the * prompt. BE SURE TO ALWAYS WRITE PROTECT THE ORIGINAL DISK WHEN YOU ARE PLAYING AROUND IN THIS MANNER. One or two minor changes by the protector and you might be off to format land.

On the above disk, when I attempted to catalog it from BASIC, I was rewarded with another]. I entered the monitor with the CALL-151 and listed from A56E and saw the following:

```
A56E: 60 RTS
A56F: 06 20 ASL 20
A571: AA TAX
A572: A2 AD LDX #AD
A574: BF B5 8D 66 LDA 668D85,X
A578: AA TAX
A579: 60 RTS
```

WOW!!! What a different program they have written into the Catalog area. Now I want you to

know that if you do NOT have a GS, and you look at this same chunk of memory; you will see an entirely different program. ????. Say What?!!

Now comes the part of this 'tutorial' that is really hard to explain on paper to different people with different degrees of understanding, but here goes. Machine language is a fascinating and wonderful communications tool. The important thing to remember is that it IS a language and just like English or any other language; the same word can have several different meanings and the placement of the word in a sentence can often change the meaning of the word. The same is true with machine language. Consider that each line of the above two listings is a sentence. If you start a sentence with 20, it needs 2 more words to complete the sentence. If you start the sentence with AA, the sentence is complete. If you start the sentence with A9, it needs one more word; but if you start with 60 the sentence is complete.

Listing machine language programs is sort of like a game we used to play when I was in elementary school English and the teacher would give us a printed sheet of paper with no capital letters and no punctuation. The trick was to know where each sentence started and work through the entire page. If you missed one sentence, it could throw the entire paper out of kilter and the results could be quite amusing.

The same is true with the second listing here. If we start the listing at A570 instead of A56E, we will get the same listing as the first except that the first 'sentence' is missing. Go back and list from A56E and look at the 'word' after A56E and we see that it is 06. It's the same second word from the first listing. Ok, it seems like the ONLY difference in the 2 listings is the very first word, 60. What if we change the 60 to A9 like in the first listing.

Type the following:

A56E: A9
A56EL

Guess what! Now it looks the same as the first listing and yes it works the same too.

Type the following:

3D0G

Now you are back in BASIC. Type CATALOG, and it even works from BASIC.

Now you know that the program actually has files that can possibly be removed. If anyone wants to see what I've been talking about, but don't have the protected program, you can reverse my last change from 60 to A9 and put 60 at A56E yourself.

Another reason why Cookbook Cracks might not work

Sometimes we see the cookbook solution of 'go to sector A on track 21 and change byte 5 from 18 to 38.' We go there and find only zeros. What gives??? One possible solution is that the protection is in a FILE on a 'normal' disk and the file is just in a different location. This is most likely why we sometimes see 'On my copy the byte was at byte 5 but on track 5 sector 7...' When writing up a softkey, if you determine that the protection is indeed in a file; mention which one. Also try to give an extra byte or two to search for if that needs to be done. Hope this is of some value to someone in the future.

Comments & possible help to other readers

Needless to say, I have gotten a bit behind in my attempt to communicate with fellow Computists. Here are some replies to letters from issues # 65-67.

To RDEX editor: How about placing any Mac softkeys you receive in their own section in the back ala IBM. That way they would be easier to find for those who care and easier to ignore by those of us who don't. Also, what happened to the subscriber number that appeared for only one issue. I for one am interested in what is going on with the fate of my magazine.

To Alan Wilson (#65, p9): I assume from your letter that you are attempting to use the old version of Print Shop with the 3.5 disks. The problem is not that the pictures are now unreadable by Print Shop; but rather, ProDOS that is unreadable by the program. If you are attempting to use old black and white graphics with Print Shop GS, then the thing to do is to convert them to GS graphics with either Print Shop GS or Graphics Exchange. The reason that you must do this is that Print Shop GS is EXPECTING to find old Print Shop components (graphics, fonts, or borders) on DOS 3.3 disks.

To Lewis D. Kaufman (#65, p27): Congratulations on your election as Apple Guru. Also Congratulations on 'solving' the unable to load... problem. There may have been an even better way to handle the situation (i.e. perhaps you could have saved the other files or at least some of them also).

First, I want to advance a possible reason as to why this problem even surfaced. This is only a guess, but the information may help some others who have the same disaster potential lurking in their world of Appleworks data. There is a MAJOR bug in all versions of Appleworks before 2.1 that will generate the type of error that you encountered if the conditions are correct.

The bug surfaces when you attempt to save a file that is already on the disk (such as a database

update with the same name) and you are greeted with the information that there is 'insufficient room on the disk for your file' and 'would you like to delete the old one?' NEVER say yes!!!

What is going on here, anyway?

Well, you know that little 'convenience' of being able to 'esc' out of a file save? That costs you something. It costs a little extra time, five spaces in your filename; and, if you are unlucky enough, it can cost you some of your files. The reason you can stop a save is that Appleworks is not saving FILENAME but, instead, FILENAME.TEMP (.TEMP is the reason you lose 5 letters in an Appleworks filename). Then if you haven't hit esc, Appleworks deletes FILENAME and renames FILENAME.TEMP, FILENAME. As you can see, when you try to save a file that already exists on the disk; you actually need room for the file twice. You can see all of this if you use a block editor on the directory of an Appleworks data disk. You can also see a little of what is happening by saving a half-dozen files to a data disk and then catalogue it with Copy II+ and note which data file is first in the directory. Re-enter Appleworks, load that file, and resave it without first deleting the old copy of the file. Then recatalog the disk and the file should now be at the end of the directory. (Don't check the disk with Appleworks because it always alphabetizes all the file types and you won't be able to tell the actual location of the file in the directory.)

Now we get to the bug. If you answer 'yes' to that fateful question, Appleworks and/or ProDOS attempts to delete the wrong file and/or saves the file incorrectly and 'trashes' some of the directory blocks. Thank you, Apple!

To avoid the bug, either use Appleworks 2.1 or newer or always answer the question with a no and go back and delete the old file from the main menu. No bug there!

Solving the problem if it has already occurred (possibly in another manner)

1. Copy the 'damaged' disk in its entirety.
2. Try to load/copy the file now. Sometimes (but not often) this will work. If it works, then immediately load all the files from the disk and place them on a newly formatted disk. Be sure to check every file as best as you can to see that it is intact and uncorrupted. Any that are not ok should be placed on a different disk for future manual repair.
3. Using your favorite block editor, try to read in every block on track 0. When you hit a stubborn one (like your sector 6), patch DOS. First try normal 'Patched DOS 3.3 with epilogs and address checksums turned off. If you still can't read the sector, patch it further by turning off the data checksum. If this works then 'unpatch DOS to normal DOS 3.3 and rewrite the sector to the copy of the bad disk.
4. Try to load/copy the files now. If you're lucky, you'll be able to save a file or two more.

A normal ProDOS directory resides on track 0 between sectors 2 and A. The reason you lost a few files was that you in effect zeroed out a block of directory by copying the empty block from Copy II+ which had less files on it. It worked because you replaced an abnormal sector with a normal one albeit blank. My above method attempts to replace the 'same' data in the sector that was there.

To Tim Furry (#65, p28): When your copy of Zany Golf was crashing, the resulting display on your screen was not 'code' but rather status information of the processor at the time of the crash. At this point, you have the 'monitor prompt', an *, and to see actual program code, you could type something like 5E00L and a page of resident memory would be displayed. What your information shows is that for some reason the program encountered a 00 (BRaK) command at the memory location 5E02 in 'main' memory while it was attempting to run. This is shown by the 00/5E02. All of the information COULD at times be important to someone attempting to softkey a program but a total explanation at this time would take up too much space.

To Rich Linville (#66, p9): If the programs that you want to transfer to 3.5 disks are really DOS 3.3 and have individual files, you might try one of the modified DOS's such as UniDOS or AmdOS to transfer the files.

Putting MECC's Oregon Trail on a 3.5 disk would be much more difficult because the DOS on the disk is very abnormal in that it contains something necessary to allow the program to run correctly. Also writing save game routines for a program such as Oregon Trail would be a major undertaking as it would be necessary to isolate all of the data (i.e. variables, program pointers etc) and save them to a file and then have another routine that would put the information back where it came from and restart the program from exactly the right location. Saving all the memory would make much more sense in this instance. If you are running the program on an IIe then it might be worthwhile to try to find an NMI board that will save all the memory and allow you to restart the program where you quit. If you are running the program on a GS then you should invest in Softswitch which has a companion feature, Keepsake, which allows you to save 8 bit pro-

grams and restart them at a later date. The only requirement is that the program must allow interrupts.

To Bud Myers (#66, p9): The GS to your rescue! I hear a lot of flack from the non-GS owners all the time about how the GS and its software don't help them out at all with their IIe's, IIc's, and Laser's. Here are two examples of the GS aiding users of the older machines.

1-GS fonts are 100% compatible with Publish It! The only difference are the file types and the fact that the GS fonts will be slightly out of proportion when printed. The file types can be changed with a block editor from C8 to F7. The file type is immediately after the file name which has 15 spaces reserved for it.

2-Graphics Exchange can be used to convert any graphic mode to any other within the limits of the resolution. For example a double hi res graphic transferred to lo res will often leave a lot to the imagination. Using a combination of my Fingerprint GSi and Graphics Exchange, almost no graphic escapes my wishes. I have converted some of the Disney Card & Party Shop 'stickers' to GS clip art and am working on converting some to Print Shop. Print Magic graphics can also be converted to a more versatile format.

To Mike Maginnis (#66, p19): The fact to note is not that Copy II+ knows when you are working with a non-original, but when you are working with a bit copy done with Copy II+. Apparently Copy II+ puts an ID byte or so somewhere on the disk to check for. According to the author of Locksmith, that program placed the serial number from the Locksmith disk on all copies that it produced. Perhaps Copy II+ does the same.

Super II Comments

When I got this issue, I thought that perhaps it was the April issue and the 'Super Computer' was another Light Simulator article. I guess that isn't the case. It seems that the author and Apple Computer must be drinking from the same well because they seem to have a lot of ideas in common. The first ridiculous idea is that the GS isn't a real II because the 'II' that is analyzed is the e/c. Apple Computer already built the 'new super computer', the IIc-!!!

I have nothing against an outside source 'improving' on the II line, but in reality a GS that just lived up to its capabilities for a more reasonable cost would be the place to begin instead of a vapor machine that costs \$2300 and has no real software base for its 'improved' features.

I feel that the author is attempting to design a Peach and call it an Apple.

To Frank Polosky (#67, p16): I am not writing this to pick on you, but hopefully to restate what the RDEX editor has said a million times. They can't print what they don't got. Perhaps you should solicit information on the nostalgic angle of gaming, compile it, and submit regular articles (kind of like a real column) like Jeff Hurlburt. Maybe even a questionnaire showing who likes what type of program/game etc. With Computist, unlike most other Computerzines, you (that is all readers) control the contents by what you do or do not send in. We can all spend pages saying what we don't like about Computist or the contributors or we can submit what we do like!

I've had some sort of an Apple since 1982 (wow!) and I can safely say that there are a lot of 'classic' games out there that do NOT belong on anyone's most wanted list for ANY reason.

I do agree 100% with you that someone out there is probably capable of writing some of those articles you mentioned and that they would be interesting and educational although personally I would prefer to turn some of the 'joystick' programs into keyboard control. The point is that with the information YOU request, I might be able to accomplish what I would like. With that in mind, try not to be so negative towards those of us who like the pretty pictures and lilting sound of the GS or the usefulness of the educational programs. Try to learn from the articles about the things that you don't like so much. You might be surprised!

To Mike Basford (#67, p22): There are times when it would be hard to implement your suggestion of using a RTS at the beginning of the subroutine because you can't tell where the beginning is without finding where it is coming from. Sometimes the code is obvious and a RTS will work just fine sometimes not. Another potential problem is that not all nibble counts etc are accessed with a JSR. Placing a 60 at the beginning of a routine may send the program into Never-Never land or worse. Another problem that can contribute to unseen problems is check-summing. Sometimes any method, NOP, RTS, BRA etc will do the job. Other times a specific cure may be called for.

That's it for this time. Thanks to all who wrote as a result of my first letters in Computist. Hope to hear from more of you.

Joseph P. Karwoski PA

I am writing to give the readers a couple of tips I have picked up in the past few weeks. My topics will be: Bit copies for two programs, copy protecting your own 5.25 ProDOS disks, and getting Appleworks to print in color.

Bitkey for...

Great Western Shootout Laser Force

Fanfare

When you try to copy these disks using the DISK/W FORMAT option in Copy II+ you get a read error on block 63F. I wanted to deprotect these programs but I didn't have the time to trace through the code, so I took a "short cut". For those of you who are interested, here is some of what I found. In Laser Force, I searched for 22 A8 00 E1 22 and I found it twice on block 4D2 (at byte 6B and D8). Also I found a C9 27 00 - bad block in the same block. In Great Western Shootout, the protection is a little different. I believe the protection is in the file STARTUP.SYSTEM, which loads at 00/0800. If you search for 14 3F 06, you will find it in one place. This is telling the program where to find the bad block. I really have not had much time to look any farther into the program. I hope someone can do a better job on these than I have done!

Now I can give you the "short cut" method of getting a working Bitcopy of each of these programs. The same method works for both programs. By-the-way, both of these programs are worth buying if you like "shoot'em up" games.

1. Copy the disk with Copy II+, using DISK/W FORMAT option.
2. When this is done, go to MANUAL BIT COPY option.
3. Set the starting block as 4F and the ending block as 4F.
4. Set sides to 2, and press return for the other options (that sets them to "N").

You now have a working copy of both games. I know this is not much help to those of you with hard drives, but it will at least give you a working back-up.

Copy Protecting Your Own Disks. ProDOS

Now I will show you an easy way of making a 5.25" protected ProDOS disk. I am only going to change the first epilog data byte. Once we are done, a normal whole disk copier will not be able to copy the disk (such as COPYA, Fast Copy, Copy II+ disk/w format option).

The steps you must follow are:

1. Format a blank disk with ProDOS. (This is as you would normally do, such as with Copy II+.)
2. Copy ProDOS, Basic.System, and all the files you want protected.
3. Catalog the disk and get the file length of ProDOS (example: 15485).
4. Boot this new disk and patch ProDOS.
CALL -151
BLOAD PRODOS, A\$2000, TSYS
53C4:97 chg ProDOS write byte from DE to 97
56C8:97 chg ProDOS read byte from DE to 97
BSAVE PRODOS, A\$2000, Lxxxx, TSYS where
xxxx is the length of ProDOS you found when you
CATALOGed the disk

This disk will no longer BOOT. You may want to keep this disk so you won't have to do this again, the next time you want to protect a disk.

5. Now we want to copy the disk to another disk so it will BOOT.

RUN COPYA

CTRL-C

70

POKE 47426,24

POKE 47262,151 chg write byte from DE to 97

RUN

You now have a disk that can not be read with a normal sector copy program. Have fun with this, but be careful not to try to write to a normal disk with this new ProDOS. If you do, it will likely make your disk unreadable.

Print in Color with Appleworks

The other day, someone ask me if there was anyway that Appleworks could print in color. using an Imagewriter II and a color ribbon. My first response was Yes, but not without some problems. The more I thought about the question, the more I thought that it could be done. I have come up with two ways of doing it. The first way is easy, but it only lets you print in one color. The second way takes a little longer to set up, but once it is done you will never have to do it again. And better yet, you can change the color you want to print with at any time (from within Appleworks).

First things first, the Imagewriter needs special-codes to print in color. they are:

Black	esc K0
Yellow	esc K1
Red	esc K2
Blue	esc K3
Orange	esc K4
Green	esc K5
Purple	esc K6

The first way (bad if you are using Appleworks) is to run a BASIC program that will set the printer to print in the color that you want. For example:

```
10 PRINT CHR$(4);"PR#1":REM Turns  
printer ON  
20 PRINT CHR$(27) + "K2":REM Sends  
esc K2 to printer
```

30 PRINT CHR\$(4);"PR#0":REM Turns
printer OFF

After you RUN this program, boot the Appleworks program (without turning the printer off at the switch). When you tell the program to PRINT, everything will be printed in RED.

The second way is to create a CUSTOM PRINTER from within Appleworks, using OTHER ACTIVITIES. I suggest that you use a back-up of your Appleworks disk to do this. The best way to get anything out of this part of the article is to sit down with Appleworks and follow the "cookbook" method. You may also want to have your printer book out to get the other codes that you MAY want to use in this custom printer. The "cookbook" method for this is as follows:

1. Get to your Appleworks' Main Menu.
2. Goto to option "5" - other activities.
3. Select option "7" - specify information
4. Take option "2" - add a printer
5. Then select option "12" - custom printer
6. Change item "3" to YES (accept top-of-page)
7. Then select item "6" - printer codes
8. Now select item "3" - boldface, subscript, superscript

Here are the setting that I used:

Boldface	Begin	escK
	End	escK0
Subscript	Begin	escYescW
	End	escZescW
Superscript	Begin	escX
	End	escY

You may set the other option to whatever you would like your printer to do. It is important that when you enter these setting that you do not put a space between anything. The computer will read the space as part of the command, and it will not work!

After you are done, press esc until you are back at the Main Menu. You now have created a custom printer that you can use whenever you want to print in color. This is a very nice way of "dressing" your letters up. Remember, Appleworks can only have 3 printers at any one time. If you already have 3 on the menu, you will have to remove one before you can add this one.

Everything will work the same except - you will not be able to boldface (this is how we will turn the color on. When you tell the computer to turn the boldface (color) on, you will have to tell it what color you want it to print by putting the number after it - this number will not be printed. The colors are:

1	yellow
2	red
3	blue
4	orange
5	green
6	purple

When you turn boldface off, you don't need to put the color in, I have done that for you. Be careful to put a valid number in when you turn boldface (color) on, or "strange" things may happen!

I know this sounds a little confusing, but if you sit down with Appleworks running on your machine, you will have no problems.

©To Dan Halfwit: In issue 68, you had a softkey for King of Chicago. It was very interesting, and gave me a few ideas for cracking Arkonoid GS (I believe the protection is very much the same on both of these programs - minor changes only). My King of Chicago had all the same bytes that you listed in your crack, and I was very pleased to find them even in the same place. I made the sector edits that you listed and tried the game. Everything went well for a short time, and then the game ended - suddenly! I found myself in the monitor with "no way" of finishing the game. My question is: Did I miss something in your softkey? I would really like to try the game, but I refuse to use a disk that I don't have a working back-up for.

David Stewart MD

Here are some collected tips. I have had them for a long while and finally decided it was time to send them in.

Playing Tip for...

Pool of Radiance

SSI

This is SSI's first official AD&D adventure. It is not copy protected but used a code wheel with lots of squiggly lines that is almost as big a pain as disk protection.

The antimagic shield around the castle of the lizard men only works partially. You may cast or memorize spells in non-combat situations. During combat, characters cannot cast spells while under your control but they can in quick mode with magic on.

To gain experience and magic items quickly for first level characters, hire Hero and go to the slums. When you get in combat, cast sleep on him and one hit with any sharp weapon will automatically kill him. Each character will get about 300 exp for this.

Multiclassed half elven Fighter-Magicuser-Clerics or Fighter-Magic User-thieves are the

best. They tend to have fairly stable hit point increases and can advance throughout the entire game, unlike single classes clerics or magic users that max out at sixth level very early in the game.

Don't use the swords in the Temple of Bane in the castle. They are cursed and will shock your characters every time you try to ready them. However, they can be sold for a substantial sum.

To avoid fighting all the buccaneers in the Slave Camp, walk around the camp until you find a man who will sell you a pass to see the captain. You can buy the boy from the captain without a fight. Better yet, break into the slave pen, steal the heir to the house of Bivant, free the animals in the corral and get out quick.

To duplicate items or money, copy your character disk, give all the money and special items to one character, and save (remove) him/her to your other character disk. One useful sector edit: To raise the maximum hit points for your character, search for his name on the save disk (Usually between sectors 3 and 5) and alter address 76 (in hexadecimal).

One last important note: It is IMPOSSIBLE to clear certain areas while on a mission. While not on a mission, it takes ten encounters to clear a sector.

Bugs in Pool of Radiance

This game was obviously rushed to the market by SSI. It has many minor and several major bugs I and my friends had to learn through experience. They are as follows:

Rings of Feather Falling do not cushion your fall but protect against poison.

Some rings of fire resistance actually function as rings of regeneration.

You cannot resurrect an elf, even though the manual does not state so.

If an unconscious character is raised a level at the training center, you can never bring him back to consciousness. If you try to cast a healing spell, the computer will state "that spell cannot help you". If the game ever says this, you must go to a temple and pay for a cure disease spell, even if your character is in good health.

If you open Cadorna's family treasure, you will not be sent on a mission to Zentil Keep.

Using the 'Area View' command on the countryside map will distort the entire screen.

If you flee from combat, all unconscious or dead characters and NPC's will be PERMANENTLY lost along with any items they may be carrying.

Unfortunately, the largest bug is fatal and ruined my enjoyment of the game. I completed all of the city clerk's commissions and wiped out all inhabitants of the castle without ever finding the bronze dragon, Tyranthraxus. The clerk ran out of commissions and the game did not say I had won although no more commissions or hints were available and I was wandering around aimlessly. Supposedly, SSI is including free hint books in all editions of Pool of Radiance starting in August (without offering any compensation to earlier buyers.)

Hopefully, the Curse of the Azure Bonds, which SSI has been procrastinating for six months now, will be better.

Playing Tip for...

Bard's Tale III

Electronic Arts

In this game, the purpose of most of the quests is to recover lost items for the Old Man so that he will tell you the location to go to to teleport to the next successive dimension. If you already know where to go to teleport, you can skip most of the quests. Table 1 should help you to avoid unnecessary treasure hunting expeditions.

You must have a Chronomancer of high enough level to cast the transport spell to enter a dimension. (Or, if you have completed the previous quest, even if you are not high enough level, the Old Man will give you the spell for free.) The return site is the place where you are transported to and it says "the ground is well worn here." Some of the dimensions are mazes and return sites have no coordinates, so be sure to note where you start when you teleport to that dimension.

Omission: To gain free experience points with my playing tips in Computist #67, you must first have killed Brilhasti ap Tarj. Once you have done this, it will function perfectly. This is the preliminary quest and almost everyone who has ever picked up the game has gotten least that far. In case you haven't, here are instructions: Go straight to Unter Brae by saying 'CHAOS' to the

priest in the temple of the mad god. Teleport 1 west, 14 north; 9 south, 12 west; and 5 south, 0 east, going down each successive set of stairs. The final (4th) level is teleport-shielded, just fight your way straight to Brilhasti's chamber, which is approx. 18 spaces north of the stairs. Kill him and you will be teleported out next to the review board, all set to exploit the game's experience point giveaway. Sorry about that.

Playing Tip for...

Wasteland

Electronic Arts

Wasteland is a neat game with an incredibly hard advancement system. My characters were still Sergeants when I won the game, and it takes 20 rank advancements just to become a Cadet! Here are some helpful hints:

There are 30 steps in the one true path.

It takes an incredibly high luck just to break even on gambling.

ACE (found in the basement of Ugly's hide-out) can fix the jeep but you need a new engine (buy one up at the Rail Nomad's camp.)

Here are the skills not listed in the manual:

IQ	Pts	Skill
19	3	Helicopter piloting
20	3	Toaster Repair
20	3	Electronics
21	3	Doctor
22	3	Cyborg Tech.
22	3	Clone Tech.
23	3	Energy Weapons

Helicopter piloting can be learned only in the simulator at Sleeper Base. Theoretically, an IQ of only 19 is needed to learn the skill, but I had to raise my character's IQ to 50 before it would even let me enter the simulator! After experimenting with other characters, I have reached the conclusion that a character must have a very high combined score of all traits to enter and can enter with an IQ of 19 only if other scores are sufficiently high.

Ⓢ After completing the game, I still have one unanswered question: how do you get the cloning machinery operational?

On a more serious note, I feel that Apple Computer is making a grave mistake in considering dropping the Apple II line. In the past, Apple has been the only major company to continue to support all of its various computers. If they go through with it and sack the II, they will be violating the trust of (hundreds of) thousands of Apple users nationwide and scare off potential buyers of the Mac as well. Apple computer is its own worst enemy; it is successfully doing what IBM and Commodore have wanted to do for years: Obliterate the Apple II line. I don't understand why Apple is so sticky with others (like Video Technology's Laser) who clone their machines that they end up voluntarily trashing anyway.

It is been my experience that only overpriced computers are cloned (ever hear of an Atari clone? or a Commodore clone?) (IBM would shoot me for saying that.) I just want to say that the new IBM is the same dumb computer it was in 1981, now it's just a "faster dumb" that has come to epitomize the very nature of bureaucratic corruption. Contrary to popular opinion, the IBM PC AT does NOT represent the best technology available by any standard. In fact, CPM was the first choice for an IBM operating system, and it would be used today if Digital had not refused IBM's offer. As far as programming goes, trace some music program on it and watch the agony that thing goes to to produce a pathetic, almost inaudible sound. And IBM machine language? HAH! right. Enough said on that.)

Also, most people probably can't afford the upgrade to a Mac or find it to specialized for them ("The Yuppy Machine") - not even color graphics in some models. The Apple is a better 'general purpose' all-around home computer that already has extensive software and hardware support as well as widespread industry acceptance, something that the Mac is still struggling with. It is almost tragic that Apple very nearly proclaims its Apple II line dead, while a comprehensive upgrade that would put the Apple II on the same level as the Mac and Amiga, is technologically quite possible and awaiting only the go-ahead order from a company official.

However, I echo Mr. Polosky's sentiment that the software available for the II just isn't good enough. Productivity software is getting more complex, but not better, so you have to spend hours reading a poorly written manual to figure out how to boot the thing up, although the

older versions and are just as functional and more reliable but without the \$200 price tag and fancy box. Like they say, if it works it's obsolete. My word processor is from '82, but it works well and I am not about to change anytime soon (If it ain't broke, don't fix it. Especially if there's nothing better on the market.) As far as the games go, the dungeons get deeper and more options exist, but much more time is required to win these latest programs, which can make them be even more monotonous than the old ones. Who really has the time to go through ten disks of screens and dungeons, and what is the point if new software offers little improvement or innovation? Software designers have yet to learn to take full advantage of the capabilities of a 128K Apple IIc or IIe. For example, Cinemaware has released all of its titles for the Commodore 64 (which belongs in a museum next to the dinosaurs and the abacus) while not one has been released for the IIe! Firebird is another big offender. On every other major computer system and even a few now-defunct ones with tape drives, their programs offer stunning graphics while the Apple versions are text only. Exceptions to this trend of poor software are few and hard to find. Take adventure games. The vast majority of them have the 'Wizardry Syndrome' where the 'plot' is a couple of one-liners scrawled on a wall at the bottom of a bottomless pit somewhere after killing endless hordes of killer mutant attack rabbits. I could name titles and companies all day, but you get the idea. I rest my case that software companies are shunning the Apple II market. This is not surprising in light of all of Apple Computer Company's neglect and many bungled opportunities.

One title I have to warn your readers about - Times of Lore. The entire game fits within 64K and retails for around \$39. That's \$.61 per K, and for a game! Get real. You can do better than that on a memory chip! If you like REAL short adventures, fine, but it's your money.

Notes on Ultima IV and V

Ultima V is an awesome game with good graphics, excellent sound (16 original songs, if you have hardware support) and an intricate plot that leaves most other games in the dust. Ultima IV is not far behind. As you play these games, remember that nothing is meaningless! In Ultima IV, each virtue associated only with one component of truth, love, or courage (Honesty, compassion, and valor) is represented by a primary color (blue = honesty, yellow = compassion, and red = valor). All other virtues have colors equivalent to the combination of all of their primary components. For example, the stone of justice, which is composed of aspects of both truth and love (blue and yellow), is green. Spirituality (with all 3 components) is white while humility (which is not composed of truth, love, or courage) is black. In Ultima V, the power words for the dungeons also have an interesting derivation: the power word is the name of the dungeon, in Latin. The word to the dungeon Doom (veramocor) is composed of the Latin root words for truth (ver - as in verity), love (amo - as in amiable or amour (french)), and courage (cor).

Ⓢ The other day I copied Infocom's Gamma Force with COPYA, which worked without displaying any errors. The backup wouldn't boot, so I ran a compare disk and changed the backup to match the original. Although the disks matched perfectly (according to Locksmith 6.0), the backup would still not boot although it got farther. Can someone give me some advice on this? (I'm sure it's something perfectly obvious, but I get hung up on these things that are second nature to most experts.)

To Rob Fduccia: In addition to making your characters fall asleep, the Chaos Sword can make the character wielding it attack others within your party and take away your control of him. The weapon is found inside Blackthorn's palace behind a number of doors that can be opened only with Skull keys. To be allowed to wander in this area, you must have the black badge (gained from Elvira in the north-easternmost isle in Britannia) and tell Blackthorn the password of the Opposition (Impera). The whole purpose of the Chaos Sword is to discourage players from politically aligning themselves with the Opposition.

To S Todd Grant: To wrap up this 'Broken Toaster' issue, there are three broken toasters I have found so far: the one you mentioned in the Quartz graveyard, one in the citadel (use pick-locking skills on one of the glass cases in the museum room to get it), and one in the middle tent in the Rail Nomad's camp (tell the guard the word 'CATERPILLAR' and then kill everyone inside and search the loot). The first toaster I fixed had a plasma coupler and some fruit inside and the second and third both had duplicates of different Citadel keys. What you do with the toaster itself I don't know.

I'd just like to say that I was glad to see a project from the hardware corner in issue #64. Keep up the good work! I hope that you have many more hardware projects planned for the future...

Suggestion: Why don't you start a separate column under the Most Wanted list for disabling documentation checks? In the words of Scott Mace (Computel editor), "...software development is supposed to entail getting away from the manual..."

Ⓢ Does anyone know of a company that still sells Sublogic's Music Maker, or have an original (with documentation) that he would be willing to trade/sell? My address is included at the end of this article so that you can write me.

Stanley Planton OH

Softkey for...

The Wonderful World of Paws

Microtype

Requirements:

COPYII+
DOS 3.3 System Master
Blank Disk

A local teacher brought in a set of disks that the kids had destroyed, and asked if there were any way to make a backup of the disks; fortunately, one of the originals would still occasionally boot, so there was something to analyze.

The program is an introduction to typing for elementary school children marketed by South-Western; it boots up with a picture of a kitten and a keyboard and displays a 1985 copyright date.

Since the program was copy protected, the initial step was to try to get it into something approximating a normal DOS 3.3 or ProDOS format. Scanning the disk's tracks with the NIBBLE EDITOR of COPYII+, I noted that the only major change from a normal track seemed to be in the address epilogs, changed to: FF FE AB on most of the tracks.

The first step in actual deprotection of the disk was to produce a relatively normal copy of the disk, so I could examine the files; since there were no drastic changes to the disk, it would be possible to use COPYA to produce a normalized copy for examination:

1. Boot the DOS 3.3 System Master disk.

2. Enter the monitor.

CALL-151

B988:18 60

B942:18

E003G

RUN COPYA

ignore address epilogs

ignore DOS read errors

return to BASIC

The result of this operation should be a copy of the disk with normal epilogs.

The next step was to check the copy to find out if it is DOS or ProDOS, or something else. Running COPYII+ showed a DOS 3.3 type of catalog, with lots of files present, including a HELLO Applesoft program, always good news, but a check of the track/sector allocation showed that track \$02 was used by files, preventing us from using a normal DOS in the softkeyed disk, since this would step all over track \$02.

A backfile of COMPUTIST issues is a very useful acquisition; I remembered that somewhere in the past there had been articles on how to make more space on a disk, and found Phil Goetz's article in COMPUTIST #30. If you don't have this issue at hand, the following is the essence of Phil's technique to free up much of track \$02 and to make a new track \$23 to hold all of these files:

3. Boot the SYSTEM MASTER

4. Enter the monitor again and make these patches:

CALL-151

B71A:00

B71E:A0 B3 EA EA

B7E0:17

B74A:A9 B4 EA

B754:A9 B3 8D F1 B7 EA EA EA EA

B763:00

B6F5:A9 FE 8D FC B3 4C FB AF

AEB3:08

AECD:F5 B6

B3EF:24

BEFE:24

AEB5:90

E003G

return to BASIC

5. Init a disk with the patched DOS.

INIT HELLO

DELETE HELLO

The next trick is to FID the files from the normalized copy of the "Paws" disk to this newly-initialized disk.

6. Boot the newly-initialized disk, then switch it to Drive #2.

7. Place the System Master disk in Drive #1.

BRUN FID

8. Place the normalized copy of "Paws" in Drive #1 and copy all of the files to the newly-initialized disk, using the = and N options when asked for file names and prompting.

8. Write protect the disk when finished and try to boot it.

You should now have an unprotected and normalized version of "The Wonderful World of Paws."

Peter Green England

Moving Destroyer GS (Epyx) to Hard Disk

This conversion will enable Destroyer GS to be placed in a sub-directory anywhere on your hard disk and is based on information from Stan Merka's article 'Putting Mean 18 on a hard disk' (Computist #68, Page 22).

Table 1

Dimension	Transport Site	Spell	Rtn Site	Spell
Arboria	Forest (7S,9W)	ARBO	4N,4E	ENIK
Gelidia	Cold Peak (7N,15W)	GELI	8N,1W	ECUL
Lucenia	Crystal Spring (2E,5N)	LUCE	5N,1W	ILEG
Kinestia	Old Dwarf Mine (9S,13W)	KINE	(maze)	OBRA
Tenebrosia	Shadow Rock (3S,4E)	OLUK	1N,3W	ECEA
Tarmitia	Vale of Lost Warriors(2S,5W)	AECE	(maze)	KULO
Malefia	Sulfer Springs (11S,3E)	EVIL	(maze)	LIVE

1. Make an unprotected copy of the original following Brian Troha's instructions in Computist #56 page 18.

2. The next stage is to modify the file pathnames within the file DESTROYER.SYS16. The easiest way is to use 'Blockwarden' but any sector editor will do.

If using Blockwarden, scan the file DESTROYER.SYS16 for the block containing a list of filenames which you will find preceded by '7'. i.e. '7/gun0' etc. On my copy, they were in the 4th block of the file. In the case of each filename (21 in all), change the '7' to '1'

If using a sector editor without a 'follow file' option, scan the disk for \$37 2F 67 75 6E 30 On my version, this string started at byte \$27 on the relevant block. Change \$37 to \$31. There are a further 20 occurrences of \$37 in this block. Change them all to \$31.

3. Create a sub directory 'DESTROYER' (or any name you like) anywhere on your hard disk.

4. Copy the files 'DESTROYER.SYS16 and 'SOUNDS' from your copy to this sub directory

5. Copy all the files in the 'ART' sub directory except 'TITLE' and 'WAVES' to the 'DESTROYER' sub directory created in step 3 above. (do NOT create an 'ART' sub directory - the 'DESTROYER' sub directory must contain all the files)

You can now play Destroyer from your hard disk. However, you will find that you do not get the title screen and music, instead the program goes straight to the screen asking for your name etc. Not being an expert with ProDOS16 etc. I have not yet been able to get the title screen and music functional. In case you are interested, you will find that the file 'SYSTEM.SETUP' in the 'SYSTEM.SETUP' sub directory on the original disk loads and displays the files 'TITLE' & 'WAVES'. This file is only run when booting up from the 3.5" disk. By converting file 'SYSTEM.SETUP' to a sys16 file, you can run the title screen and music, but with my lack of knowledge, I have been unable to make this file then run the 'DESTROYER.SYS16' file. Any ideas?

Running other programs from your Hard Disk

I use a Cirtech SCSI card with my hard disk. The software supplied with this card enables you to set up partitions intended for use with Pascal, etc. However, if you create a Pascal partition of the appropriate size, you can then copy all the files (including the system files) from the program disk you wish to place on the hard disk. If you then rename the partition file from 'PASCAL' to a name of your choice, you can repeat the operation for the next program disk. I have copied such programs as Marble Madness, Silent Service, etc. to my hard disk using this method with no problems. Remember however, if you wish to delete a partition, you MUST rename it back to 'PASCAL' and use the Cirtech software to remove it. If you delete the partition file without using the Cirtech software, the blocks allocated for the partition will not be marked as free on the hard disk!

Softkey for...

Super Print

Ashton Scholastic

The softkey for Super Print (COMPUTIST #71, pg13) does not work. Scanning the file revealed a further 4 indirect jumps to the reboot routine mentioned and NOPing them eliminates the need for part of the softkey that Robert Phillis was puzzled about. I suspect they may have been accidentally omitted from the article.

The simplest way to deprotect this program is:

1. Boot ProDOS and get into BASIC.

PREFIX /SP/

UNLOCK SP.SYSTEM

BLOAD SP.SYSTEM, A\$2000, TSYS

CALL-151

2026:EA EA EA

2033:EA EA EA

2064:EA EA EA

20AD:EA EA

20B4:EA EA

26B0:EA EA

26B7:EA EA

26CF:EA EA

26D6:EA EA

BSAVE SP.SYSTEM, TSYS, A\$2000, L1929

2. Get back into BASIC and lock SP.SYSTEM.

Ronald E. Riggs

NV

Notes on Time Out and Appleworks 2.1

I had been using AppleWorks V-1.3 for several years. Admiring the power that the Beagle Bros "Time Out" series of AppleWorks add-ons had to offer. I decided to upgrade to V-2.0. I purchased the new version of AppleWorks through the mail from one of the discount software places. I then purchased Beagle Bros. Superfonts and Desk Tools. When I tried to install them I found they were not compatible with my copy of AppleWorks. You see, I had

received V-2.1. I wrote a letter to Beagle Bros. who informed me that these programs were indeed not compatible with V.2.1 of AppleWorks and it would cost me an additional \$10 to upgrade each program. Still wanting the power offered by Beagle Bros. I later purchased Time Out Power Pack. It is compatible with AppleWorks V-2.1. When I had it up and running I thought I would copy the applications from Desk Tools and Super Fonts to the Power Pack applications disk. Low and behold, the programs worked. I now have all the power of the programs I had purchased with out having to part with an additional \$20.

All you have to do is copy all of the programs with the letters "TO" on the front of their file names to your Time Out applications disk.

No Name

CA

Softkey for...

Algebra 1

Algebra 2

Algebra 3

Eduware

Guinness World Records

Addition & Subtraction

SVE

Meteor Multiplication

Decimal Discovery

Freddy's Puzzling Adventures

DLM

Spelltronics

Math for Everyday Living

Educational Activities

Mixed Numbers

Reading Comprehension

Pick the Numbers

D.C.Heath/MB

Balance

HRM

Galaxy Math Games

Peanuts Math Matcher

Charlie Brown's 1,2,3's

Mr. and Mrs. Patatohead

The Boars Tell Time

The Boars Store *

CBS's Pathwords

Random House

Math Tutor (Percents

Ratios & Propotions

Fractions

Whole Number Operations

Decimals)

Microzines and Microzine Jr.

Scholastic

Esimation

Fraction Recognition

Percentages

Edusoft

Galaxy Search

Magic Word

Learning Well

Ivitation to Math series

Scott Forsman

Opposite

Numbers

Alphabet

Townbuilder

Stickybear

Early Games

Springboard

Spelling Bee

Eduware

(*Note: Copy a new ProDOS)

These softkeys are also dedicated in fond memory of Roger Trapp whose help and friendship I miss

and who it seems could unlock any piece of software given time.

These softkeys were developed with the help of other readers who have sent in their modifications to COPYA from the DOS 3.3 System Master.

All of these programs copy with the following:

RUN COPYA

ctrl reset

CALL-151

B988:18 60

B98A:00

B925:18 60

B942:18

At the slot prompt

3DOG RUN COPYA

Return to BASIC

With some of the programs you have to use something like Copy II+ to see what the BOOT program is then copy a normal DOS to the disk and RENAME your BOOT program as the original was names.

Softkey for...

World Geograph

MECC

Use the same softkey as Calendar Crafter in Computist #62.

Wanted a softkey for:

Graphic Converter, Pelican

Geometry, Broderbund

Loderunner, Broderbund (#28's doesn't work on mine)

Choplifter, Broderbund

Mr. Robot, Broderbund (#24's doesn't work on mine)

Gary O

MO

Softkey for...

The Duel:Test Drive II

Accolade

I found a way of cracking this using information found in Mr. Brian Troha's softkey for Graphics Studio from Computist #62, page 25.

I searched for the bytes A2 20 and found them in block 16E. Changing them to 80 2C gets a copy that doesn't lock up when going into the game portion of the program.

1. Copy the original The Duel:Test Drive II disk.

2. Make the following edits to the copy:

Blk	Byte	From	To
16E	134	A2 20	80 2C

3. Write the block back to the copy.

Mark Provence

CA

Softkey for...

The Games: Winter Edition

Epyx

Requirements:

COPYA

Sector Editor

3 blank disks

This softkey is based on all the other issues about the Prolok protection scheme, but specifically from issue #64. I have found that a good rule of thumb on programs released around the same time, and being the same type (i.e. Summer Edition and Winter Edition), will have the same protection scheme (or at least pretty close). As with the Summer Edition, only side 1 is protected. I searched for the big segment of code on a normalized copy, and it all was there, byte for byte. Then, with close examination of what I had on my screen, and the softkey from #64, I determined that the D0 xx bytes were the last 4 sets before the large code. With that in mind, I changed those bytes, and, Viola! Everything works.

1. Boot DOS 3.3

POKE 47426,24

RUN COPYA

ignore checksum & epilogs errors on side 1 only

2. Make these sector edits:

Trk	Sct	Byte	From	To
00	05	43-44	D0 F4	EA EA
		4C-4D	D0 4A	EA EA
		55-56	D0 41	EA EA
		69-6A	D0 F4	EA EA
		6C-79	A0 07 BD 8C C0	EA EA A9 FC 85
			10 FB 99 F0 00	F0 85 F3 A9 EE
			EA 88 10 F4	85 F1 85 F2

Note: if the sector edits are not there use a search utility and search for A0 07 BD 8C etc. Things should then look VERY familiar.

Now use any other disk copier and copy the remaining sides. I wish though there would be some way to eliminate the disk "grinding" on boot-up.

Softkey for...

Math in a Nutshell

Learning Technology's

Another mini-softkey - I deprotected this one with only the swap controller. It came up as a DOS 3.3 disk with an "]" cursor on bootup so the swap is the first thing I try. Maybe most of their software will succumb to it??

Here is a little tip for the beginners that, to some, might be too obvious. But I believe there will be some who haven't given the possibility any thought. For some of those binary utilities create a Basic loader file that has the instructions built in. Lets take the RWTS worm program that came out a while back. I love the program but I may not have my COMPUTISTs with me when I crack something or I can't remember where the issue is that has the instructions in how to use it (I know look at the back issue log in the COMPUTISTs and find it that way, hang in there). The issue said Bload the routine to a specific area in memory, insert disk to capture, make the call, insert DOS 3.3 disk, Bsave RWTS and Swap

away. Simply make a basic program with nothing more than print statements explaining just that and the last line is the load statement (PRINT CHR\$(4)"BLOAD RWTS.WORM, A\$....."). The only suggestion I have for loader files is make the load statement last. Sometimes these writers load something into memory that the print statement may overwrite accidentally. For the most part the writers do not but it is just a good rule of thumb to use. NEVER assume.

On the BBS idea, if I wasn't so far away I'd volunteer to help run it. I help run 2 GBBS boards in our area (Riverside County area in California).

Please keep Strike Fleet on the most wanted list I am stumped on that one like everyone else and now is a good time to purchase it since Electronic Arts decided to place the program in the "classics" category.

Michael Doucette

NH

Advanced Playin Technique for...

Tetris

Spectrum Holobyte

The original game of Tetris has a protection scheme that requires you to type in a password referenced in the manual by page, line, word. This was a real pain, because after you know how to play the game, you shouldn't have to keep the manual out. At the direction of a friend, I searched the text for a password that I had used and found that all the passwords and their page, line, word, equivalents were in order at block \$0100 and \$0101. They are case sensitive when you type them in. And here they are:

Page	Line	Word	Password
4	12	8	leaves
4	15	11	layers
4	21	6	options
4	24	1	grasp
5	4	4	original
5	9	4	ready
5	11	5	already
5	13	13	create
5	22	3	System
6	5	11	example
6	7	6	locate
6	16	6	music
6	21	4	panel
7	1	8	number
7	5	4	toggles
7	11	4	Height
7	18	3	pieces
7	19	1	previous
8	1	10	button
8	4	6	numeric
9	6	8	fashion
9	25	4	select
9	31	2	reappear
9	34	1	highest
10	3	3	command
10	11	10	Comrades
10	12	2	selected
10	17	7	music
10	19	12	assist
11	5	6	total
11	10	8	level
11	13	4	feature

Jack Nissel

PA

Softkey for...

Square Pairs

Scholastic

Requirements:

A blank disk

DOS 3.3 system disk

A sector editor

1. Boot your DOS 3.3 system disk and at the

Applesoft prompt enter:

POKE 47426,24 ignore checksum and epilogs

RUN COPYA

2. Copy the original disk to your blank.

3. Make the following sector edits to your copy.

Trk	Sct	Byte	From	To
00	05	93-95	C6 2A D0	4C 86 02

4. Write the sector back to the disk.

Softkey for...

Microzine #3

Microzine #9

Microzine #11

Scholastic

Requirements:

2 blank disks for each title

DOS 3.3 system disk

A sector editor

1. Boot your DOS 3.3 system disk and at the

Applesoft prompt enter:

POKE 47426,24 ignore checksum and epilogs

RUN COPYA

2. Copy the original disk to your blank.

3. Make the following sector edits to your copy.

Trk	Sct	Byte	From	To
00	05	93-95	C6 2A D0	4C 86 02

4. Write each sector back to the disk before going to the next sector.

Softkey for...

Microzine #8 Microzine #10 Microzine #12 Microzine #13 Microzine #15 Microzine #16

Scholastic

Requirements:

2 blank disks for each title
DOS 3.3 system disk
A sector editor

1. Boot your DOS 3.3 system disk and at the Applesoft prompt enter:
POKE 47426,24 ignore checksum and epilogues
RUN COPYA
2. Copy the original disk to your blank.
3. Make the following sector edits to your copy.

Trk	Sci	Byte	From	To
00	05	93-95	C6 2A D0	4C 86 02

4. Write the sector back to the disk.

Softkey for...

Microzine #22

Scholastic

Requirements:

3 blank sides
DOS 3.3 system disk
Copy][Plus

1. Boot your DOS 3.3 system disk and at the Applesoft prompt enter:
POKE 47426,24 ignore checksum and epilogues
RUN COPYA
2. Copy both sides of the original double-sided disk to 2 of your blank disks.
3. Boot Copy][Plus and use the COPY DOS option to copy the DOS from your DOS 3.3 system disk, (or any normal DOS 3.3 disk), to side 1 of your copy.
4. Use the Copy][Plus CHANGE BOOT PROGRAM option to check, and if necessary change, your boot program on side 1 to HELLO.
5. Use the Copy][Plus COPY DISK option to copy your original single sided disk to your last blank disk. No changes are necessary to this disk.

Softkey for...

What makes a Dinosaur sore

D.C. Heath and Company

Requirements:

2 blank disks
DOS 3.3 system disk
Any file copy program
Any ProDOS 8 program with a normal ProDOS file

1. Boot your DOS 3.3 system disk and at the Applesoft prompt enter:
POKE 47426,24 ignore checksum and epilogues
RUN COPYA
2. Copy the original disk to your blank.
3. Use your file copy program to delete the ProDOS file from your copy and then copy the normal ProDOS file to your copy.

Softkey for...

Teasers by Tobbs Memory Castle

Sunburst Communications

Requirements:

Blank initialized disk(s) for each title
A blank initialized slave disk
Super IOB v1.5
A way to reset into the monitor

To deprotect these titles we will use Super IOB with the Swap Controller and use the RWTS of the protected disk to read the original disk then use a normal RWTS to write the information back to your blank disk.

1. Initialize your blank disk(s).
INIT LOGO
DELETE LOGO
2. Boot your original disk and at the Applesoft prompt reset into the monitor.
3. Move the RWTS to a safe place by entering:
1900<B800.BFFFM
4. Put your slave disk in the drive and boot it:
C600G
5. After the disk boots and the Applesoft prompt appears insert your Super IOB disk and save the RWTS to it by entering:
BSAVE RWTS.SUNBURST,AS1900,LS800
6. Install the controller into Super IOB, run it and copy your original disk to your blank disk. Answer NO when asked if you want to Initialize the blank disk.

Controller

1000 REM SUNBURST
1010 TK = 3:ST = 0:LT = 35:CD = WR
1020 T1 = TK:GOSUB 490:GOSUB
360:ONERR GOTO 550

```

1030 GOSUB 430:GOSUB 100:ST = ST
+ 1:IF ST < DOS THEN 1030
1040 IF BF THEN 1060
1050 ST = 0:TK = TK + 1:IF TK <
LT THEN 1030
1060 GOSUB 490:TK = T1:ST = 0:
GOSUB 360
1070 GOSUB 430:GOSUB 100:ST = ST
+ 1:IF ST < DOS THEN 1070
1080 ST = 0:TK = TK + 1:IF BF = 0
AND TK < LT THEN 1070
1090 IF TK < LT THEN 10201100 HOME
:PRINT "COPYDONE":END
10010 IF PEEK (6400) < > 162 THEN
PRINT CHR$ (4)
"BL0AD0RWTS.SUNBURST,AS1900

```

Checksums

1000-\$356B	1040-\$6342	1080-\$6CA2
1010-\$3565	1050-\$ABA3	1090-\$496E
1020-\$6170	1060-\$20C0	10010-\$4FEA
1030-\$7771	1070-\$28C5	

An APT or a Softkey?

In answer to the question as to whether or not an edit to remove password/codewheel/picture protection should be called a softkey or an APT. An APT is something that gives you an advantage when playing something. For example, unlimited lives, more power, more weapons, etc. A patch that removes the password protection does not give you any type of advantage in the game.

If you are going to call the patch an APT just because the disk is already COPYA-able then what about patches that remove the nibble count check on disks that are also COPYA-able?

To me a softkey is something that removes copy protection no matter what form it is in. Maybe you should revise the meaning to reflect that.

Bob Dylan

OR

Softkey for...

Children's Writing & Publishing Center (3.5")

The Learning Company

I actually stumbled onto this one. I searched for the message to insert the original, and then found some suspicious code nearby. Make the following changes:

Blk	Byte	From	To
\$3DA	\$04D	B0 01	60 60

Softkey for...

Geometry (GS)

Broderbund

This program comes with 3-3.5" disks. The first is a system disk, and the other two contain the actual program. Both disks are protected using nibble counts. I searched for A2 20 A0 01 and found this on disk 3, block 7 (located at 01/176F in memory). I looked for jumps to this routine, and was able to track it back a couple of steps in memory. The same protection is used on both disks, but the code is found in different locations for each. Make the following changes:

Blk	Byte	From	To
\$408	093	22 8E 0D 00	EA EA EA EA

Blk	Byte	From	To
\$E7	1CE	22 8E 0D 00	EA EA EA EA

Softkey for...

Word Attack Plus! Spanish

Davidson & Associates

This is a ProDOS8 program, with a block check. Search for 20 00 BF 80 and change the 20 to 60. On a 5.25" disk, it should be on track \$6, sector \$A.

Rex Creekmur

MI

Softkey for...

APBA Major League Players Baseball

Random House

Requirements:

Bit Copier
Sector Editor Disk Copier
Game set consisting of Master Program, Play By Play Disk, Draft Disk and Data Disk(s)

As a new subscriber to Computist, one of the first things I did was scan the back issues for long-sought softkeys. I found one in Issue #59 from Mike Egnotovich for a program I have been able to only partially deprotect and copy, APBA Major League Players Baseball. However, Mike's softkey is incomplete. Here, as Paul Harvey says, is the rest of the story.

1. Make a bit copy of the original Master Program. I used Essential Data Duplicator 4 with the original drive set to 300 rpm and the duplicate drive set to 297 rpm. Recopy any tracks that show errors.
2. Using a sector editor (Copy II Plus), make the following edit on the copy:

Trk	Sci	Byte	From	To
\$21	\$00	\$C2	10 FB C9 D5 F0 F4 D0 F0	EA EA EA EA EA EA EA

The Master Program Disk is now deprotected and can be copied using a normal whole disk copier. You'll get an error on Track \$21, but it doesn't matter since the nibble count routine has been bypassed.

3. Make a bit copy of the Draft Disk.
4. Use the sector editor to make the following edit on the copy of the Draft Disk:

Trk	Sci	Byte	From	To
\$12	\$0D	\$C2	10 FB C9 D5 F0 F4 D0 F0	EA EA EA EA EA EA EA

5. Make a normal disk copy of the Play By Play Disk. (You can make a bit copy if you prefer, but it's not necessary.)

6. Use the whole disk copier to copy both sides of any Data Disks you plan to use. DO NOT BIT COPY! The Draft Disk has to write to the Data Disks and you will get an "UNRECOVERABLE SYSTEM ERROR" if you bit copy them.

You now have a complete set of deprotected disks that can be backed up by normal means. Thanks again to Mike Egnotovich for his original work in writing the softkey for this program. See you at the ballpark!

Does anyone have a softkey for Mi-croLeague Baseball or a doc protection bypass for Earl Weaver Baseball?

Kris Kirk

AK

Playing Tip for...

Robocop

Data East

Robocop is a great arcade game but one you have to play a long time to master (if your able to master it). One night while I was getting beat up pretty badly I started messing with the keyboard and found these great commands hidden in the program:

- ctrl A Will allow you to fire single round shots. The screen shows that your magazine is "FULL", I've never been able to empty it. It's probably not possible.
- ctrl B Will allow you to fire armor piercing bullets. You are reset to 25 rounds each time you press this one.
- ctrl C Will allow you to fire three-way bullets. You are reset to 15 rounds each time you press this one.
- ctrl D Will give you a weapon that shoots balls of fire, it's called a "Cobra Gun". You are reset to 15 rounds each time you press this one.
- ctrl E This is the best one. This one stops the clock and makes you immortal. That's right, you can't die.
- ctrl F This one reverses ctrl E. It starts the clock back up and you become a mere mortal again.
- ctrl J Puts you in joystick mode.
- ctrl K Puts you in keyboard mode.
- ctrl R Takes away hit points. Once all your hit points are gone you die. This one doesn't work while your in the ctrl E mode.
- ctrl S Toggles sound ON/OFF.
- esc Pauses game.

Using these commands I was able to make it to the last level. I couldn't have made it any other way. Just too many things shooting at you.

Comments

The format you have now selected for Computist is fine with me. It wouldn't matter if you printed it on toast I would still be waiting with bated breath every 6 weeks for my loaf to arrive in the mail.

I would like for you to put Robocop on the Most Wanted List. It's a great game and I don't want to lose it. Has anyone out there got a crack for Tetris, the //e version.

I would be glad to help anyone out there who is having trouble with Appleworks(tm) and the Timeout(tm) series of enhancements. I feel that I've experienced about every possible thing that could go wrong, and that I am quite proficient with Appleworks(tm) and Timeout(tm). Also, if anyone out there likes to use Ultramacros(tm) and you have some unique macros you wouldn't mind sharing drop me a line. I'm a macro nut from way back and I'm always looking for new ones.

Richard S. Thompson

CA

Softkey for...

Test Drive II: The Duel (IIGs)

Accolade

This is the first IIGs program that I have deprotected. Take a look at this disassembly on block \$16E:

128:08	PHP	
129:C2 30	REP #30	set 16-bit registers
128:5A	PHY	
12C:8B	PHB	
12D:4B	PHK	
12E:AB	PLB	

12F:9C 97 C5	STZ	\$C597	init pass value ???
132:E2 30	SEP	#30	Go into 8-bit registers
134:A2 20	LDX	#20	Load track
136:A0 01	LDY	#01	Load side
138:20 40 C5	JSR	\$C540	Do nibble count
13B:8E 97 C5	STX	\$C597	
13E:8C 98 C5	STY	\$C598	
141:C2 30	REP	#30	Go into 16-bit registers
143:AD 97 C5	LDA	\$C597	
146:C9 6C 20	CMP	#206C	
149:90 29	BCC	\$174	Branch if not original
14B:C9 02 21	CMP	#2102	
14E:B0 24	BCS	\$174	Branch if not original
150:E2 30	SEP	#30	Go into 8-bit registers
152:A2 21	LDX	#21	Load track
154:A0 01	LDY	#01	Load side
156:20 40 C5	JSR	\$C540	Do nibble count
159:8E 97 C5	STX	\$C597	
15C:8C 98 C5	STY	\$C598	
15F:C3 30	REP	#30	Go into 16-bit registers
161:AD 97 C5	LDA	\$C597	
164:C9 80 1D	CMP	#1DB0	
167:90 0B	BCC	\$174	Branch if not original
169:C9 78 1E	CMP	#1E78	
16B:B0 06	BCS	\$174	Branch if not original
16E:C2 30	REP	#30	Go into 16-bit registers
170:AB	PLB		
171:7A	PLY		
172:28	PLP		
173:5B	RTL		
174:C2 30	REP	#30	Go into 16-bit registers
176:A9 00 D0	LDA	#D000	
179:8F AB 35 02	STA	\$0235AB	Mess up A-register (I think?)
17D:AB	PLB		
17E:7A	PLY		
17F:28	PLP		
180:6B	RTL		

If the program makes it to \$16E, it is an original. So I put a BRA \$3C at \$132 to bypass the protection scheme all together.

1. Fast copy the master disk.
2. Change these bytes on the disk:

Blk	Byte	From	To
\$16E	\$132	E2 30	80 3C

That's it. If you don't find the bytes at this location, search the disk for E2 30 A2 20 and change E2 30 to 80 3C. The best thing to do now is use Mr. Fixit on this disk, because there are 20 blocks marked used that are not. Mr. Fixit is a disk repair program from ProSEL. And then use Beach Comber on it. (Also from ProSEL)

(?)I have tried to put GS/OS 4.0 on it but every time it boots it comes up with a \$201 error. Does anyone know what a \$201 error is?

Softkey for...

Where in the World is Carmen SanDeigo? (GS)

Broderbund

This softkey took me three weeks. I have to thank Joseph P. Karwoski for the excellent article on deprotecting software. I used his 'bomb' (68 FA 00) and it worked perfectly. Thanks. I was ready to pull my hair out, but I finally figured it out. I have a short explanation for this softkey because I do not know what this program is doing exactly, but I did figure out enough of it.

On Disk 2, Block \$96, Byte \$44, you should see the following:

44:A9 00 00	LDA	#0000
47:85 F2	STA	\$F2
49:64 F0	STZ	\$F0
4B:A5 EE	LDA	\$EE
4D:F0 03	BEQ	\$52
4F:82 A0 00	BRL	\$1F2

So, all I did was put two NOP's at \$4D. It is now deprotected.

1. Fast copy both disks.
2. Change these bytes on Disk #2:

Blk	Byte	From	To
\$96	\$4D	F0 03	EA EA

As with above, if you don't find the bytes at this location, search the disk.

Sidney P. Bobe

CA

Bug in Type! softkey

In Computist #60, pg22 the From/To bytes appear to be reversed. I found EA EA EA and changed it to 20 00 01 and all is readable now.

Fred Gluck

CO

Is there anyone that has successfully managed to run Applewriter IIe on a hard disk? In two tries to load and use AWIIe on my sider harddrive, I have clobbered track 00 to the point where I had to reformat the drive. I can't be sure that AW is the culprit but the problem started shortly after I started experimenting with it.

An earlier issue of COMPUTIST stated that AW's DOS is essentially the same as DOS 3.3 but loaded in a different place. Given that, it is hardly surprising that it doesn't work with the hard drive.

Anyway, if someone has any helpful ideas, I would appreciate them. I've also written to 1st Class Peripherals about this problem.

Scumbucket

IBM Softkey for...

Shinobi

?

To unprotect the game Shinobi (with the file SH.EXE dated 9/23/89) make a backup copy of your disks or copy them to a hard drive. Then do the following:

```
REN SH.EXE SH.E
DEBUG SH.E
E 6C7
xxxx:06C7 9A.EB 00.03
E 6D1
xxxx:06D1 75.EB
W
Q
REN SH.E SH.EXE
```

George Aram NY

I've added an IBM to the 4 Apples in my house. I have an IBM program, "Pirates!" that I want to put on the hard drive. Does anyone know how to do this?

Charles S Taylor CA

Bug in APT for Zany Golf

There is a bug or misprint in Marc Batchelor's APT for Zany Golf on the IBM. The line reading: S100 FFFF 8A 84 FA 59 3C 01 F5

```
should read:
S100 FFFF 8A 84 FA 59 3C 01 F5
```

Otherwise, the crack and APT work perfectly and are a real improvement in the game.

The following softkeys were sent to us on disk. They are from a BBS. RDEXed

Brian H. Lawler

Softkey for...

688 Attack Sub

Electronic Arts

This patch will relieve you of the ludicrous hassle of searching the manual before every mission. I feel that this form of copy protection is as much a pain in the rear as any other.

```
REN GAME.EXE GAME.BIN
DEBUG GAME.BIN
E 606C EB 1F
```

```
W
Q
REN GAME.BIN GAME.EXE
```

When the screen prompts for the letters just leave them as AAA and press the Send button.

Softkey for...

Battlehawks 1942

Electronic Arts

Patch to run without trying to identify those stupid planes.

```
REN BH.EXE BH.BIN
DEBUG BH.BIN
E 2D56 2E
```

```
W
Q
REN BH.BIN BH.EXE
```

When the screen prompts for the code name just press RETURN.

Note: that the screen will print that the code failed and that you are doomed to "duck shooting" - press any key and you will proceed to the selected mission.

Ralph and the Computer Pike

Softkey for...

Chuck Yeager's Advanced Flight Simulator v1.0

Electronic Arts

Requirements:
A blank formatted diskette
The program DEBUG.COM (found in most MS-DOS packages)

Overview

The copy protection for Chuck Yeager's Advanced Flight Simulator (by Electronic Arts) was very simple, so save the \$10 for the unprotected version offered by Electronic Arts. Tracing was done on the file AFS.EXE using Microsoft's Symbolic Debug Utility - SYMDEB.COM (version 3.00). Concentrating on any disk I/O, it was found that a portion of the program had four identical groups of two CALLs bunched together. The first CALL routine was CALL 0B31, which resets the disk, reads the boot record and track 15 (side 0). By the way, track 15 (side 0) is one of those "hard to copy" tracks. The second CALL routine was CALL 0B74, which checks the status of the flags set by the last disk I/O, being CALL 0B31. Replacing these CALLs with NOPs (three

NOPs per CALL), we found that the program ran with no problems whatsoever! It was that easy! As a matter of fact, we joked about the fact of merely using NOPs to unprotect the program before running it... It couldn't be that easy! Sure enough...

Procedure

In copying the program diskette, we decided use COPY *.*. Reasons being that it allows the COPIED diskette to rid the bad track 15 (side 0). It was found that using DISKCOPY returned errors and COPYIIPC left track 15 (side 0) in its "bad" state. So, let's do it this way... If all goes well, you'll only have to do it once! Perform the following:

1. Assuming that the PC has been booted-up and the DOS prompt is B>, insert the ORIGINAL program diskette into drive A: and the blank FORMATTED diskette into drive B:.
2. Type the following after each B> prompt:
MD AFS
COPY/V A:.*.*
COPY/V A:\AFS.* B:\AFS
3. Now, remove the ORIGINAL program diskette from drive A: and put it away... You won't be needing it anymore!
4. Insert the diskette containing the DEBUG.COM file into drive A:.
5. You'll have to rename the AFS.EXE file. Type the following at the B> prompt:
RENAME \AFS\AFS.EXE AFS.XXX
6. Now, enter debug by typing the following at the B> prompt:
A:DEBUG \AFS\AFS.XXX
7. At each hyphen "-" prompt, type the following:
E DE0 90 90 90 90 90 90
E DE8 90 90 90 90 90 90
E DF1 90 90 90 90 90 90
E DF9 90 90 90 90 90 90
W
Q
8. You'll have to rename AFS.XXX back to its original name. Type the following:
RENAME \AFS\AFS.XXX AFS.EXE

There! You now have an unprotected copy of Chuck Yeager's Advanced Flight Simulator by Electronic Arts...

Conclusion

Being an avid user of PC-based flight simulators, I'll have to say that Chuck Yeager's Advanced Flight Simulator ranks among the BEST! It's a shame that many of you frowned upon the copy protection... We hope that this document has aided in removing such a nuisance, and that it will confidently allow you to use the software freely. Have fun and good luck!!!

Addendum - For single-drive users

First of all, I'd like to apologize to all you single-drive users. I don't think you all could have done the above verbatim. So, I went ahead and COPYIIPC'd (version 3.09) the ORIGINAL program diskette in drive A: Ignoring the three errors that track 15 created, I went ahead and used the copy and performed the unprotection. I ran the copy with no problem. Then, I DISKCOPY'd (PC-DOS 3.2) the diskette in drive A: Again, I ignored the errors. The DISKCOPY'd diskette also ran with no problem. Unless you have a program that corrects or formats by tracks and sides, you'll have to live with the errors popping up during copying with no effect to the actual running of the program. Such a program that allows formatting by track and side is called ULTRAFMT.COM. It is a public domain program and usually comes as a package with other ULTRA programs. The programs can be found on most BBSes under the ARC'd filename ULTRA.ARC. Again, good luck and... sorry!!!

Buckaroo Banzai - the Reset Vector

Cracking On the IBMpc

Many copy protection schemes use INT-13 either to try to read in an illegally formatted track/sector or to write/format a track/sector that has been damaged in some way. It is called like any normal interrupt with the assembler command INT 13 (CD 13). [AH] is used to select which command is to be used, with most of the other registers used for data.

Although INT-13 is used in almost all protection schemes, the easiest to crack is the DOS file. Now the protected program might use INT-13 to load some other data from a normal track/sector on a disk, so it is important to determine which tracks/sectors are important to the protection scheme. I have found the best way to do this is to use Locksmith/pc to analyze the diskette. Write down any track/sector that seems abnormal. Load the file into DEBUG and execute a search for CD 13. Record any address shown. If no address is picked up, the program is not copy protected or the check is in another part of the program not yet loaded (a real bear to find). Another choice is that the CD 13 might be hidden in self changing code like this:
U CS:0000
1B00:0000 31DB XOR BX,BX
1B00:0002 8EDB MOV DS,BX

```
1B00:0004 BB0D00 MOV BX,000D
1B00:0007 8A07 MOV AL,[BX]
1B00:0009 3412 XOR AL,12
1B00:000B 8807 MOV [BX],AL
1B00:000D DF13 FIST WORD...
```

In this section of code, [AL] is set to DF at location 1B00:0007. When you XOR DF and 12, you would get a CD(hex) for the INT opcode which is placed right next to a 13 ie, giving you CD13 or INT-13. This type of code can not and will not be found using debug's Ssearch command.

My favorite way to find hidden INT-13s, is to use PC-WATCH (or TRAP13). These trap the interrupts and will print where they were called from. Then you can disassemble around the address until you find code that looks like it is setting up the disk interrupt. Another way to decode the INT-13 is to use debug's G) command. Just set a breakpoint at the address given by PC-WATCH (both programs give the return address), ie, -G CS:000F (see code above). When debug stops, you will have decoded not only the INT-13 but anything else leading up to it.

Once you find the INT-13, all you do is fool the computer in to thinking the protection has been found. To find out what the computer is looking for, examine the code right after the INT-13. Look for any branches having to do with the Carry Flag or any CMP to the AH register.

If a JNE or JC (etc) occurs, then U)assemble the address listed with the jump. If it is a CMP then just read on. Here you must decide if the program was looking for a protected track or just a normal track. If it has a CMP AH,0 and it has read in a protected track, it can be assumed that it was looking to see if the program had successfully completed the Read/Format of that track and that the disk had been copied thus JMPing back to DOS (usually). If this is the case, Just NOP the bytes for the CMP and the corresponding JMP. If the program just checked for the carry flag to be set, and it isn't, then the program usually assumes that the disk has been copied. Examine the following code

```
INT 13 Read in the Sector
JC 1B00 Protection found
INT 19 Reboot
1B00 (rest of program)
```

The program carries out the INT and finds an error (the illegally formatted sector) so the carry flag is set. The computer, at the next instruction, sees that the carry flag is set and knows the protection has not been breached. In this case, to fool the computer, just change the "JC 1B00" to a "JMP 1B00" thus defeating the protection scheme.

Note: The protection routine might be found in more than one part of the program

Debug reads .EXE files but can't write them. When the protection scheme has been found and tested, record (use the debug D)ump command) + & - 10 bytes of the code around the INT 13. Exit back to dos, rename the file anything but .EXE and reload with debug. Search the program for the 20+ bytes surrounding the code and record the address found. Then just load this section and edit it like normal. Save the file and exit back to dos. Rename it back to the .EXE file and it should be cracked.

Note: Sometimes you have to mess around for a while.

Disk I/O (INT-13)

The INT-13 interrupt uses the AH register to select the function to be used. Here is a description of the interrupt.

- AH=0 Reset Disk
- AH=1 Read the Status of the Disk system in to AL. AL=Error# (* denotes most used in copy protection)
00 - Successful
01 - Bad command given to INT
*02 - Address mark not found
03 - write attempted on write prot
*04 - request sector not found
08 - DMA overrun
09 - attempt to cross DMA boundary
*10 - bad CRC on disk read
20 - controller has failed
40 - seek operation failed
80 - attachment failed
- AH=2 Read Sectors
- input
DL = Drive number (0-3)
DH = Head number (0 or 1)
CH = Track number
CL = Sector number
AL = # of sectors to read
ES:BX = load address
- output
AH = error number (see above) [Carry Flag Set]
AL = # of sectors read
AH = 3 Write (params. as above)
AH = 4 Verify (params. as above -ES:BX)
AH = 5 Format (params. as above -CL,AL ES:BX points to format Table)

If you can't find CD 13 in the .EXE file:
1 - the .EXE (usually .COM) file is just a loader for the main file

- 2 - the .EXE file loads in an overlay
- 3 - the CD 13 is encrypted &/or hidden in the .EXE file or
- 4 - you are looking at the WRONG file.

I won't discuss case 1 because so many UNP files are devoted to PROLOCK and SOFTGUARD and you can figure it out with them. If you have case 3, use the technique above and restart from the beginning. If you have case 4, shoot yourself.

You know the program uses overlays but don't see any on disk? Try Nortons; any hidden files are probably the overlays we are after. If you still can't find them, use PC-WATCH (a must!! Traps ALL interrupts). Start up PC-Watch and EXCLUDE everything in the left col. Search the right col. until you find DOS21 - OpnFile and select it. Now run the program to be cracked. Play the game until the protection is checked. Examine your pwatch output to see what file was loaded right before it. This probably is the one with the check. If not, go through all files.

When you find the overlay just crack it as if it was a DOS file. Keep a backup copy of the overlay so if you mess up, and you will, you can recover quickly. Locksmith's (AlphaLogic) analyzing ability is great for determining what and where the protection is. I find it useful, before I even start cracking, to analyze the protected disk to find and id it's protection. This helps in 2 ways. First, it helps you to know what to do in order to fake out the protection. Second, it helps you to find what the program is looking for. Check your local pirate board for the program. I also suggest getting PC-Watch and Norton Utilities 3.1.

UNK

Softkey for...

Battle Chess

?

```
REN CHESS.EXE CHESS.$$$
DEBUG CHESS.$$$
E 72FD 90 90 90
E 7303 31 C0 90
E 730C EB 53 90
E 7377 EB 14 90
W
Q
REN CHESS.$$$ CHESS.EXE
```

If that doesn't work then use PCTOOLS or other such program and search CHESS.EXE for:

Search for	Replace with
74 18 FF 46 F2	EB 18 FF 46 F2
E9 98 FC B8 77 00	90 90 90 B8 77 00
75 09 C4 5E F4	90 90 C4 5E F4
74 04 EB FE EB 59	EB 04 EB FE EB 59

That's it. Now at question time, just hit enter.

Wanted

IBM RDEX Editor

I'm looking for an IBM RDEX editor, someone to help edit the IBM submissions. I'll forward material to you for basic editing. You must have and use an IBM or compatible. Since I don't, it's very hard to format programs and source listings for printing as text.

Dave Goforth is helping with the IBM editing but he's already busy with the BBS. We could use a helping hand.

If you are interested, let me know. RDEXed

unClassifieds

How to place an UnClassified Ad

If possible, send text on a 5 1/4 inch Apple formatted disk, include a typed sample copy with appropriate instructions. Use up to 40 characters per line, we will adjust word wrap. The Computist club member charge is \$4 (for processing) plus 50 cents per line. For non-members, the charge is \$4 plus \$1 per line. Multiple insertions of the same ad are charged only for the line rate, unless changes are made to the copy.

Special Graphics Instructions: The first three words of the first line are printed in bold for free. If you want other words bolded, use 5 characters less per line. Use 10 characters less per line if you have a lot of uppercase bold letters. Bold letters are wider than normal. Circle the words you want bolded. If you want a line centered, write CENTER next to that line. There is no charge for centering any line.

You must check your ad for errors, the first time it runs. Errors on our part will be corrected, then, for free. Errors or changes on your part will be charged the \$4 processing fee.

Our liability for errors or omissions is limited to the cost of the ad.

We reserve the right to refuse any ad.

Washington state residents add 7.8% sales tax.

Send a check or money order (funds drawn on US bank only) for the entire amount to:

COMPUTIST unCLASSIFIEDS
33821 East Orville Road
Eatonville, WA 98328

unClassifieds

RENT IIGS SOFTWARE

- * Rent 99% of Commercial Software
- * 60 Day Money Back Guarantee
- * As Low as 10% of List Price
- * Pay By Week, the Choice is Yours
- * \$15 Lifetime Membership Fee
- * START SAVING NOW !!!!

GSoft

7350 Ulmerton Road #924
Largo, Florida 34641

DID YOU KNOW?

The **COMPUTIST SUPER INDEX** has over 1000 softkeys and tips not printed in the contents or back-issue listings of Computist. With issue #71, the **CSI** data base has over 5,000 records and 26,000 entries! Includes Apple, IBM, & Mac. Order today! See back cover of issue #66 or 72, or write to:

David R. Hopkins
3495 W. Hoye Place
Denver, CO 80219.

Computist Issues for Sale. Complete set of Computist-Hardcore—1 to present (minus issue 7) \$150.00. Senior PROM IIe - \$100.00.

James Heil
3809 Cynthia Dt
Pittsburgh, PA 15227

TRADE YOUR APPLE SOFTWARE

Send your list of programs to trade. I have over 120 originals to trade.

Byron Blystone
PO Box 1313
Snohomish, WA 98290

RDEX Contributors:

George Aram	38
"Scotty" B.M.E. Upp	24
June Baker	14
Jim Bancroft	14
Daniel Bashford	19, 23
Marc Batchelor	9
Bad Bob	27
Sidney P. Bobe	37
Buckaroo Banzai (the Reset Vector)	38
B. Dudley Brett	20
Richard Brooks	11
Doyle H. Brown	20
Gary Chan Keng Hwee	20
Chaos	31
Rex Creekmur	37
Datablaster	24
Gustavo Delfino	26
B. Walter Disney	22
Michael Doucette	36
Bob Dylan	19, 37
Mr. E	26
Edison	10
Mike Egnotovich	28

Fred Gluck	37
David Goforth	4, 12
Peter Green	35
Dave Grenda	15
Maverick Game Hacker	15
Jim S. Hart	9
James J. Harvey	18, 24
Brian Hatch	18
James Hodge	28
David R. Hopkins	15
Jeff Hurlburt	7
Bob Igo	25
John Jackson	26
Bill Jetzer	17, 28
Blain Johnson	24
Raymond Karl	27
Joseph P. Karwoski	34
Dennis Kathrens	24
Kris Kirk	37
Zeno Kopecky	28
Robin Locksley	18
Sandy Lapp	28
Dick Lavalley	20
Brian Lawler	38
Mike Maginnis	25
Troi Miller	26
Jack Moravetz	19
No Name	36
The Nipper	20
Jack Nissel	36
Gary O	36
Charles V. Perrien	12
Ralph & the Computer Pike	38
Stanley Planton	35
Philip C. Plunkett	13
Mark Provence	36
A Loyal Reader	14
Keith B. Reed	23
Ronald E. Riggs	36
Ro	30
Jeff Root	11
Mark J. Ruskin	10
George Sabeh	23
Scumbucket	38
David Stewart	34
Charles S. Taylor	38
Richard S. Thompson	37
Bill Todd	18
UNK	38
Roger Williams	33
Everett B. Young	20
Zorro	15

Great Western Shootout (IIGs)	Britannica
Gutenberg Sr.	Micromation LTD.
Halls of Montezuma	Electronic Arts
Hard Hat Mack	?
High Orbit	Softsmith
Horizon V	Softsmith
Impossible Mission	Epyx
Indoor Sports	Mindscape
Infocomics	Infocom
Jack Nicholson: Greatest 18 Holes	Accolade
Jane	?
Joker Poker	Mindscape
Kabul Spy	Sirius
Keyboarding Klass	Mastery Development
Kingdom of Facts	Santa Barbara/Thunder Mountain
Lane Mastodon	Infocom
Lancaster	SVS
Laser Force (IIGs)	Britannica
Legacy of the Ancients	?
Lost Tomb	Datasoft
Manhunter New York IIGs	Sierra On Line
Mavis Beacon Teaches Typing (gs)	Software Toolworks
Microwave	Cavalier
Might and Magic II	Activision
Minotaur	Sirius
Modem MGR	MGR Software
Mr. Pixel's Cartoon Kit	Mindscape/Thunder Mountain
National Inspire	Tom Snyder Productions
Neuromancer (IIGs)	Interplay
Observatory (The)	Mindscape/Lightspeed Software
Odin	Odessta
Operation Wolf	Taito
Pensate	Datasoft/Softdisk
Phantasie II	SSI
Phantoms 5	Sirius
Pig Pen	Datamost
Project: Space Station	Avantage
Pulsar II	Sirius
Pure Stat Basketball	?
Qix	Taito
Quadratic Equations II	Olympus Educational Software
Questron II	Electronic Arts
Rails West	SSI
Rastan	Taito
Rear Guard	Adventure International
Renegade	Taito
Rescue Raiders	Sir Tech
Rings of Saturn	Level 10 ?
Risk	Leisure Games
Rocket Ranger (IIGs)	Cinemaware
Roundabout	Datamost
S.D.I. (IIGs)	Cinemaware
S.E.U.I.S.	SSI
Sea Stalker	Broderbund
Serpentine	Broderbund
Slipheed (IIGs)	Sierra
Skeletal System	Brainbank
Sky Shark	Taito
Sound Song & Vision	Advanced Software
Space Ark	Datamost
Spare Change	Broderbund
Spectre	Datamost
Speedy Spides	Readers Digest
Star Cruiser	Sirius
Star Maze	Sir Tech
StickyBear Math: Add & Subtract	Optimum Resources
Stickybear GS Versions 3.5	Xerox
Strike Fleet	Electronic Arts
Succession	Piccadilly
Superstar Ice Hockey/Mindscape	Mindscape
Superstar Indoor Sports	Scholastic
Talking Text Writer GS	Origin Systems
Tangled Tales	Spectrum Holobyte
Tetris (IIG)	Cinemaware
The Three Stooges (IIGs)	PBI
Theatre Europe	?
Thunder Chopper	Blue Lion Software
Ticket to Washington D.C.	Electronic Arts
Tomahawk	Datasoft
Tomahawk (IIGs)	Broderbund
Track Attack	Adventure International/Thunder Mountain
Triad	California Dreams
Triango (IIGs)	Infocom
Trinity	Origin Systems
Ultima Trilogy	Earthware Comp. Services
Volcanoes v1.8	Melbourne
War in the Middle Earth	Electronic Arts
Wasteland	Sirius
Wayout	Broderbund
Wings of Fury	Sir-Tech.
Wizardry: Return of Werda	Davidson
Word Attack Plus (IIGs)	First Star Software
Works (the)	Softsmith
Zenith	Infocom
ZorkQuest	Infocom

Most Wanted

50 Mission Crush	SSI
Airheart	Broderbund
Alcon	Taito
Alien Mind	PBI Software
Ankh	Datamost
Apple Panic	Broderbund
Aquatron	Sierra
Axis Assassin	?
Bad Street Brawler	Mindscape
Bank Street Writer Plus	Broderbund
Beyond Zork	Infocom
Bilestoad	Datamost
Blue Powder - Grey Smoke	Grade
Border Zone	Infocom
Borg	Sirius
Bouncing Kamungas	Penguin
Boxing	?
Bubble Bobble	Taito
Bureaucracy	Infocom
Caverns of Callisto	Origin
Centauri Alliance	Broderbund
Checker	Odessta
Chess 7.0	Odessta
Chuck Yeager's Advanced Flight Trainer	Electronic Arts
C'est La Vie	Adventure International
Comics	Accolade
Cosmic Relief	Datasoft
Crime & Punishment	Imagic
Crossword Magic v4.0	?
Cybernation	Nexa Corp.
Cytron Masters	SSI
Deathlord	Electronic Arts
Delta Squadron	Nexa Corp.
Desecration	Mind Games
Disk Optimizer System	Nibble Notch
Dondra	Spectrum Holobyte
Dragon Eye	Epyx
Dueling Digits	Broderbund
Dungeons and Dragons	Master Assistant vol2
Dungeon Master (IIGs)	FTL
DROL	Broderbund
Epidemic	SSI
Epoch	Sirius
Explore-Australia	Dataflow Computer Service
Evolution	Sydney
Falcons	Piccadilly
Factastics Trivia	Daystar
Fit Wars	Sirius
Force 7	Datasoft
Gamma Force	Infocom
Gemstone Healer	SSI
GEOS	Berkley Softworks
Gertrudes Puzzles	?
Galactic Gladiators	SSI
Gladiator	Taito
Goldrush	Sierra On Line
Gorgon	Sirius
GradeBuster 1 2 3	Grade Buster

IBM Most Wanted

GBA Championship Football	Electronic Arts
Graphitti	George Best Phillips Acadmey
Gunship	Microprose
Heros of the Lance	SSI
Kings Quest III	Sierra
Operation Wolf	Taito
Radio Baseball	Electronic Arts
Ultima V	Origin

Find it Fast! Make it Easy!

Looking for a **quick and easy** way to find your favorite softkey, article or play tip without having to re-read your entire COMPUTIST library? This is what you are looking for:

The Computist* Super Index!

The Computist Super Index is a data base listing of all the Softkeys, Articles, Play Tips, APT's, and RDEX inputs printed in Computist magazine. Three colossal indexes with over 5,000 records and 27,000 entries! (Over 100 records and 500 entries added each issue!)

With The Computist Super Index, you instantly have the title, issue, page, special requirements, bugs, and "type" of softkey! At the touch of a key, you can search, sort, and print softkeys by program name, publisher, or de-protection method! By entering a key word or characters, the location of every article, tip, and entry about your topic of interest, is immediately sent to the screen or printer.

Extra Bonus! The Computist Super Index includes over 1,000 softkeys and tips (hidden within the RDEX), not printed in the contents or back-issue listings of Computist! If it is printed in Computist, it is listed in The Computist Super Index! (IBM and MacIntosh entries included.)

*David Hopkins is not connected with COMPUTIST magazine and *The Computist Super Index* is not a product of COMPUTIST or SoftKey Publishing.

Features

- Three (index) data bases: Softkeys, Inputs-Articles-Programs, and Play Tips-Advanced Playing Techniques-Reviews.
- Over ten fields to search and sort, including: Title, Issue, Page, Distributor, Input Location, Softkey Type, Bugs, Author, and more!
- All RDEX entries are summarized and categorized! Categories include: Articles, Programs, Editorials, Queries, Inputs, Listings, Help Wanted's, and others.
- The Computist Super Index lists virtually everything printed in Computist, from Hardcore

Volume 1, Number 1, to the present! IBM entries in separate indexes.

- Softkey "Type" identifies the exact deprotection techniques used for each softkey! Includes: Sector & Block Edits, Modified Copya, Controller, DOS Patch, File Copy, Bootcode Trace, RWTS Capture, Hex Dumps, Hardware Copy, etc.
- Format: Appleworks, DIF, or Text (ASCII). Please note: Since Appleworks screen and printer formats are lost when the files are converted to DIF or Text (ASCII), Appleworks is the recommended format.

Order Form

Computist Super Index

The Computist Super Index is updated each issue. All orders sent Air Mail. Price includes shipping. Hardware requirements: Apple II, II+, IIe, IIc, IIgs, or compatible, with at least 128K RAM for the Appleworks version, 48K for the DIF and Text versions. Not available in MS-DOS format. The indexes are broken into 40K segments for the 5.25" version and RAM size less than 256K. Order The Computist Super Index today!

Computist Super INDEX (5.25" or 3.5") U.S. \$10.00 Foreign \$12.00

Update Subscriptions

An Update is a wise investment! An Update Subscription is even better!

- (1) Each update contains the entire data base, plus the current issue, and all corrections and revisions.
- (2) You get a new, fully updated, copy of The Computist Super Index with each update, and at half of the original index price!
- (3) Each update totally replaces all previous versions!
- (4) Save money, time, and effort with update subscriptions. The more updates ordered, the greater the savings!

An update subscription is four to eight updates, sent every one to four issues. Please note: You must purchase the INDEX before you can get an update subscription.

How to order:

- (1) Copy order form.
- (2) Mark desired options.
- (3) Complete questions.
- (4) Send with check or money order to:

David R. Hopkins
3495 W. Hoye Place
Denver, CO 80219

What COMPUTERS are you using? _____
What is your RAM SIZE? _____
What DATA BASES are you using? _____
What WORD PROCESSORS are you using? _____
What ISSUE # do you want to start with? _____

Thank you for your order and support!

	Every issue	One every 2 issues	One every 3 issues	One every 4 issues
U.S. (5.25" or 3.5"):	<input type="checkbox"/> \$36.20	<input type="checkbox"/> \$28.20	<input type="checkbox"/> \$20.20	<input type="checkbox"/> \$21.20
Foreign (5.25" or 3.5"):	<input type="checkbox"/> \$49.40	<input type="checkbox"/> \$38.10	<input type="checkbox"/> \$26.80	<input type="checkbox"/> \$27.80
Total Updates:	8	6	4	4

Please specify format, DOS, and disk size by checking one box in each row.

Format: Appleworks Text (ASCII) DIF
DOS: ProDOS DOS 3.3 GS/OS
Disk: 5.25" 3.5"

Name _____

Address _____

City _____ State _____ Zip _____

• Send U.S. funds drawn on U.S. bank. Not available by charge card or COD. Prices subject to change without notice. Please make checks payable to: David R. Hopkins