When you're in the <u>Know</u> you'll <u>Know</u>:
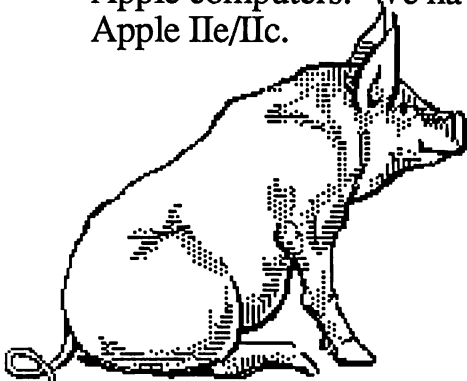
Williams COMPUTER CENTER

We at Williams Computer Center feel you should be informed. As you know the computer industry does change on a daily basis and especially for Apple Computer. Our staff here at WCC are dedicated hard working individuals specializing in different areas of expertise. Mr. Williams, (Jerry)owner - technician, has a broad knowledge of all the Apple products from the III's to the latest Macintosh Plus. Sharon Williams and Cathy Psencik work with education, from school districts to hands-on in our classroom. Betty Sigsbee handles all our in house details like accounting, personnel and business accounts. Larry Davis, formally of Apple Computer, is our sales manager and works with the larger corporations. Steven Bass handles predominantly Macintosh solutions. If your concern is with engineering or CAD-CAM he may be able to offer a plan of action. Our head technician Doug Taylor, (it is said), can fix anything. We as a **team** service what we sell and fix it while you wait. If its not broken there is no charge. Since our people do not work on commission, we feel you are served better. At the same time our business can maintain that special harmony that we've grown to love. We feel confident in saying we are the only **"Apple Only Store"** in the south Texas-Lousiana area.

Items	Retail Price	Club Price
Scribe Ribbons	$4.99	$4.39
Apple Hard Drive 20	$1199.00	$1079.10
Data Frame 20	$1095.00	$985.50
Hyper Drive 10	$1395.00	$1095.00
AppleWorks with SpellWorks	$299.95	$250.00
Mac+ Carry Bags	$79.95	$71.96
ImageWriter II Bags	$45.00	$40.50
Ramfacter 256K	$239.00	$215.10
Inside Macintosh (hard cover)	$79.95	$70.36
Quark 20 (IIc-IIe-Mac-Mac+)	$1595.00	$1095.00
LightSpeed "C" for Mac.	$175.00	$157.50
Macintosh 800K Ext. Drive	$399.00	$359.10
Paradise 20MB	$1595.00	$1095.00

We have a complete line of parts instock to fix your hardware while you wait. Also in our show room we have over 300 titles on display expanding the spectrum of Apple computers. We have over nine hard drives for the Apple Macintosh and Apple IIe/IIc.
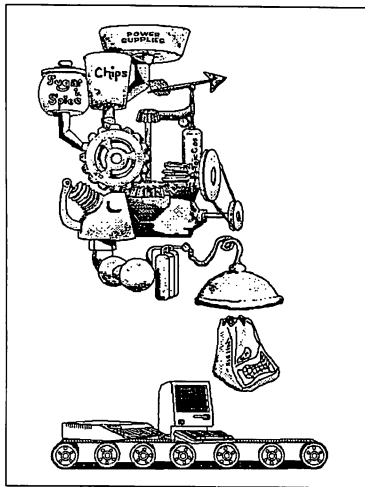
Williams COMPUTER CENTER

25770 I-45 North
Spring, Texas 77386

713-367-1600

WCC Texas Talker BBS
24 Hours 713-0367-8206

About the Cover

A fanciful vision of the Apple production line graces our cover this month. Hidden in the picture are the initials of the artist.

Contents

Overview

The Houston Area Apple Users Group provides a forum for the exchange of ideas and information about Apple Computer products and their associated hardware and software. Meetings, newsletters, and special interest groups are just a few of the ways this forum is maintained. Below are listings of the club officials and descriptions of the groups' functions.

Club Officials

President
Robin Cox — 778-1635
First Vice President
Tom Dillon — 376-6502
Second Vice President
Michael Conway — 495-2292
Secretary
John Marek — 782-3898
Treasurer
Jack Cowart — 467-4215
Directors
Steve Bass — 847-4407
Jeanne Boucher — 723-9519
Dick Fairman — 723-9619
Mike Kramer — 358-6687
Dick Lee — 821-1298
Bill Muhlhausen — 668-3963
Senior Advisor
Rudge Allen — 622-3979
Membership
Neal Scott — 890-0532
Apple Barrel **Editors**
Tom Engle — 774-9055
Chris Flick — 363-3153
Publicity Coordinator
Dennis Bilbe — HotLine
IAC Representative
Ina Levinson — 270-0137
Vendor Coordinator
Jeanne Boucher — 723-9519
Program Coordinators
Mike Kramer — 358-6687
Steve Bass — 847-4407

The club officials meet on the first Thursday of the month at the Marriott Astrodome Hotel at 7 pm. These meetings are open to the membership. Members interested in the inner workings of the group are encouraged to attend.

Membership Information

A one-year membership to HAAUG is available for thirty dollars and includes the Starter Kit, a subscription to the *Apple Barrel,* and a membership card giving access to all the club's services. Memberships are available at the Answer Desk during the meeting, through brochures located at local dealers, or by calling the HAAUG HotLine and having a brochure mailed to you. Renewals are twenty dollars per year.

Meeting Dates

The HAAUG monthly meetings take place at the Marriott-Astrodome Hotel, 2100 South Braeswood at Greenbriar. The meetings are scheduled for the third Saturday of the month and begin at 11 am. All new members should attend the New Member Orientation.

Meeting Schedule

11:00	New Member Orientation
12:00	HAAUG Club Business
12:30	Main Presentations
1:30	Special Interest Groups
2:30	Special Interest Groups
3:30	Special Interest Groups

HAAUG HotLine (713) 522-2179

The HAAUG Hot Line provides a means for **members** to learn of meeting topics, news, etc, and to obtain answers to computer-related questions. If you want a return call, leave your name, telephone number, and membership number. If the question pertains to a particular commercial product, please check the list of volunteer specialists in this issue.

Special Interest Groups

SIGs focus on one topic of common interest during these meetings within a meeting. Meeting locations and times are contained in the meeting handout.

SIG Leaders

Apple // SIG Coordinator
Tom Dillon — 376-6502
Appleworks-Beginning
John Slack — 491-1747
AppleWorks-Intermediate
George Pierce — 526-5103
Advanced Topics
Bill Hensley — 980-4993
Jim Record — 353-1119
Basic Programming-Beginning
Robin Cox — 778-1635
Basic Programming-Advanced
Richard Lemay — 749-3135
CP/M
Kip Reiner — 463-7074
Education
Ann Petrillo — 489-7535
Games
Thomas Carson — 861-5425
John Newell — 531-9139
Graphics
Dick Fairman — 723-9619
Green Apples
Richard Goss — 463-0640
Hardware
Mark Stevens — 341-8197
Home Use
Mike Stoops
David Jaschke — 461-2450

Mac SIG Coordinator
Steve Bass — 847-4407
Communications
Mike Conway — 495-2292
Desktop Publishing
Cleland Early — 941-7247
Tom Engle — 774-9055
Developers
John Pence — TBA
Games
Bob Lewis — 242-0990
Helix
Larry Stage — 358-1105
Omnis
Buddy Jacks — 299-1555
Programming
Mike Martin — 333-6405
Tips & Techniques
Steve Bass — 847-4407

Public Domain Software Libraries

The Public Domain Libraries are divided into two sections: Macintosh and Apple //.

Macintosh Library

The Macintosh library contains over ninety disks of public domain software. Copies may be purchased at the meeting for five dollars apiece. In addition, there are two directory disks which may be copied for free. These disks contain the listings of what is on the other disks. Also, the latest disk submissions are available for copying.

Macintosh Librarian
Larry Saphier   496-0507

Apple // Library

The Apple // library may be copied at the meeting for one dollar per side. Books are available at the meeting describing what is available in the library.

Apple // Librarian
John Marek   782-3898

Hardcopy Library

The Hardcopy Library maintains an archive of back issues of the *Apple Barrel*, issues of other magazines, and other paper material. It is open to the membership and is available at the main meeting. At present, no provisions for loaning the material have been made.

Hardcopy Librarian
Les Brown   795-2741

"HAAUG Heaven" Bulletin Board System

HAAUG Heaven is HAAUG's computerized Bulletin Board System. The BBS's telephone number is (713) 664-9873 and operates 24 hours a day. The recommended protocal setting is 8-N-1-Full.

Writer's Guidelines

The HAAUG *Apple Barrel* exists to provide the membership with timely information about their Apple computers. We at the *Barrel* hope that readers will write articles on subjects such as reviews, computer techniques, and casual observations of computerdom. For those of you interested in writing for the *Barrel* (hopefully most of you), we have established helpful guidelines to follow when writing and submitting articles.

1. Try to include some form of graphic or table which helps to explain points made in the article. The saying that a picture is worth a thousand words certainly applies here.

2. For those including graphics and tables, please make references to them within your text. This will help both the reader understand what you are saying and assist us in the paste-up of the article.

3. Include references to other sources of information if possible. By doing so, the reader is directed to other useful material beyond that of the actual text.

4. For Apple // users, we will accept text created either in *AppleWriter* or in the *AppleWorks* word processor. We prefer ProDOS, but can handle DOS 3.3 as well.

5. For Macintosh users, we accept text created in either *MacWrite* or *Word*. Please do not include graphics within the actual text. Graphics accompanying the articles should be saved as *MacPaint*, *MacDraw*, or *Chart* files.

6. Articles should be provided on disk as well as printed hardcopy. The hardcopy is primarily used to allow us to see where you as the author place emphasis. (A backup copy for yourself would be advisable)

7. Article submissions can be given to either of the editors or left at the Apple Answer Desk. Include both your name and phone number with the article to help us communicate with you if problems arise.

We hope that these guidelines will not only help you but also make it easier for us to provide a more consistent and classy newsletter.

Advertising

Rates

Inside Covers	$100
Full Page	$80
Half Page	$60
Third Page	$40
Business Card	$15
Other	$5 per column inch

Ads are to be prepaid, on disk, and received by the third Saturday of the month before publication. Contact either of the Editors for further information.

Random Thoughts

My Struggles With *MacPascal*

It seems like such a nice, elegant concept. Write code for the most common software operations, stick it on a silicon chip, and thus make it available for all programmers to use, so that all programs developed for the computer will have a common appearance. However, after six or eight months of intermittent fooling around with programming on my Macintosh, I've discovered that getting all that nice prewritten code to work in my programs is not so easy. I can understand why commercial software developers initially struggled so much.

I've been writing my programs in *Macintosh Pascal* (lately in version 2.0 — a major improvement over version 1.0). *MacPascal* was not designed to be a real development language, but I can't imagine why the designers, Think Technologies, thought anyone would want to write a Mac program without resources, menus, dialog boxes, and the other elements which make the Mac user interface so appealing. *MacPascal* 2.0 supports resources, and you can use inline commands to call most of the toolbox routines. The problem is, Apple provides very few instructions on how to use the inline commands with *MacPascal*. In other words, be prepared to use the programmers' (reset) switch a lot.

Apple does provide three manuals with *MacPascal*, and the disks contain a small file on inlines. The slim *User's Guide* is fine as far as it goes, but the *User's Guide* does little more than describe the *MacPascal* menus and windows. To really begin programming, you'll need to read the *Reference Manual* and the *Technical Appendix*, both of which are uniformly terrible. I guess they are clear to the guy who wrote them, but I think a novice programmer would find both the *Reference Manual* and the *Technical Appendix* daunting; I know I do, and *MacPascal* is the third language I've explored.

The quality of the *MacPascal* manuals and other documents is not atypical. On the contrary, all (at least all I've seen) computer language manuals are the pits. That's why there's such a big market for supplemental language books. If the manuals that came with the software were any good, you wouldn't need *Programming in So-and-so*. I've also found that one supplemental text is usually not enough. You see, the supplemental texts are generally not all that hot either. I usually turn from book to book trying to find an explanation that makes sense.

Suddenly, a lot of our cultural heroes are businessmen. What's happening here?

I have found one excellent book that deals almost exclusively with *MacPascal* inline commands. *Mastering the Macintosh Toolbox* by David B. Peatroy & DATATECH Publications has understandable prose, good example programs, and a terrific reference section. Peatroy's book is not exhaustive, however; there are many toolbox routines he doesn't cover. Stephen Chernicoff's *Macintosh Revealed* (volumes I and II) is not as accessible, but it is more complete. Unfortunately, Chernicoff's examples are all written in *Lisa Pascal*, so the procedures and functions he describes don't work exactly the same way in *MacPascal*.

My other supplemental books are not as good. *Macintosh Pascal* by Moll and Folsom was commissioned by Think Technologies as the tutorial which should have come with the program itself. I find most of the book's example programs unnecessarily trivial, and the book doesn't even mention the toolbox or inline commands. *Games and Utilities for the Macintosh* by Dan Shafer contains 27 example programs written in *MacPascal*. I can't vouch for the programs' usefulness, for though I've entered four or five of them, I've yet to get any of them to work even though I've checked them line by line. Example programs that don't work aren't much use.

In all, I'm pretty frustrated by the lack of adequate documentation for *MacPascal*. I think you ought to be able to write a procedure that works the way the books say it should. I think you ought to be able to create a program without so much diddling and guesswork. I don't think the programmer should be faced with cryptic error messages that raise more questions than they answer. Maybe that's unreasonable.

But I am making progress. Pascal itself does not strike me as that much different from COBOL and BASIC, the languages with which I am already familiar. All three are procedural languages, and like most COBOL nowadays, Pascal is a structured language. I do find the Macintosh concept of pointers and handles difficult. I also think the rigorous type-checking Pascal performs is a pain. Nevertheless, I've learned how to create menus, and even how to put icons in those menus. Now, I'm able to write a prototype program which calls a set of dummy procedures.

The idea is to write and debug a real procedure and substitute it for one of the dummies. Sadly, I'm stuck on windows. I've learned how to create windows, but the program crashes every time I try to move one of them. I can't figure it out, and my books are little help. I shouldn't be surprised, really; I've encountered similar problems in

other settings. Alas, I've come to expect better things from Macintosh software — it's generally superior to its IBM or Apple II counterpart. Macintosh languages, it seems to me, shouldn't be any different.

Worshipping Business

I just about puke every time I hear or read someone say that Apple has to please the Fortune 500 in order to survive. I mean, we're talking about Shylock, the moneychangers, the philistines; yet, apparently inexorably, business users are becoming the focus of the microcomputer industry.

If you doubt that, take a look at *Personal Computing* or *InfoWorld*. *Personal Computing*, a magazine that never was very good, is now even worse. The July issue contains business-oriented articles on (1) saving money with speedier communications, (2) writing faster, and (3) making a salesforce more productive. Dull. The magazine is filled with pictures of guys in coats and ties and with business buzzwords like "payback period." Deadly dull.

InfoWorld, which used to be my favorite computer publication, no longer contains John C. Dvorak's entertaining column. Instead, you'll find a column called "The Corporate View." You'll find a column headlined "Employee Needs Must Determine Leadership" by Ken Blanchard, the author of *The One Minute Manager*. Fascinating stuff. *InfoWorld* used to be lively and opinionated; now it's starting to get awfully dull.

Suddenly, a lot of our cultural heroes are businessmen. What's happening here? We used to make fun of the guy who said "What's good for General Motors is good for the country." Now there are people (Democrats!) who want to run Lee Chrysler for president. College business schools are producing more business students than businesses can possibly absorb. Business how-to-books are bestsellers, even though the thoughts expressed in most of these books are startlingly unremarkable.

I'm sorry, but I think there's something wrong with a society which venerates business tycoons and neglects artists, writers, scientists, architects, and engineers. These are the real creators of society. Businessmen are great at counting beans and exploiting others' ideas, but there's little real creativity at work there. Don't get me wrong. Greed has a place in society, and we need guys to count the beans, but businessmen shouldn't be heroes, and commerce shouldn't be the highest calling for our citizens. Necessary evils, yes.

I fear that Apple's current management will make a pact with the devil in order to sell more computers to the Fortune 500 companies when they'd be wiser to approach the mass-purchasing, big companies obliquely. Apple has reportedly let a contract to develop the BIOS for a IBM compatible coprocessor board for the new slotted Macintosh. IBM PC software? We're talking dull here. Apple is trying to be more market- rather than technology-driven, meaning they'll design products to meet the desires of their customers. Certainly Fortune 500 executives have the bucks to spend, but I doubt they have the wisdom to "design" new and exciting computers. There's danger in a company being too market-driven.

Market-Driven?

While on the subject, the dichotomy between market-driven and technology-driven companies strikes me as a little bit bogus. Most successful companies, with the possible exception of IBM, have succeeded by being both technology- *and* market-driven. By definition, a technology-driven company develops new products first, then seeks a market for the product. A market-driven company, in contrast, finds what its customers need (or want) and then develops products to meet those needs. Apple Computer has been roundly criticized for being too technology-driven, while IBM is generally considered the quintessential marketing company.

If you think about it, IBM has never really been known for stunning technical innovation. For example, the IBM PC is little more than an enhanced version of the Apple II — slightly better processor, more memory, detached keyboard. Instead, IBM's trademark has been its remarkable, Machiavellian marketing techniques. College mar-

keting professors will tell you that marketing is different from selling (market-driven companies market; technology-driven companies sell), but IBM's forte has always been selling, particularly to unimaginative commercial customers.

If you think about it, IBM has never really been known for stunning technical innovation.

In her book, *The Computer Establishment*, Katherine Davis Fishman describes how IBM came to dominate the computer industry. In the mid-fifties, Sperry Univac held a technological lead over IBM, but Sperry took the wrong approach in selling. "Univac salesmen," Fishman writes, "well-primed in the special features of their equipment, visited company presidents and talked learnedly of dual circuitry, metal tape and Mercury delay lines; IBM salesmen promised that their computer would get the payroll out two days early and save vast sums of money in the process; and there was no question whose argument was more persuasive."

Fishman also suggests that, later, IBM exploited its momentum and great wealth to maintain its number-one position. IBM engineers did a lot of R&D, but IBM executives actually retarded technological innovation in order to keep their leased equipment in the field longer and increase IBM's return on investment. When a competitor, one of the BUNCH (Burroughs, Univac, NCR, Control Data, Honeywell), introduced a new product, IBM, its hand forced, would trot out a model incrementally better than the competitor's new one, and orders for the new BUNCH computer would dry up. IBM's approach, though effective, is hardly admirable.

You can bet that John Sculley lives in fear of IBM co-opting the Macintosh by creating a Macintosh-like computer. That's probably why Apple has increased their R&D budget so dramatically in the last year; why they

bought a Cray supercomputer to model computer architectures; and why they plan to hire 100 more engineers to work on new products. Apple plans to battle IBM by continuing to be innovative, and I like that. I am, however, worried that the kind of innovations we're likely to see from Apple in the future may be the incremental type favored by Big Blue rather than the stunning improvement represented by the Lisa, Macintosh, and LaserWriter.

Computer companies have long known the way to compete successfully with IBM is to carve out a marketing niche. A couple of examples: Cray Research dominates the supercomputer market; Digital Equipment Corporation has succeeded by selling commodity-like VAX minicomputers to scientific research institutions. In the commercial arena, Burroughs has survived by selling tailored systems to banking institutions.

Apple has now identified three areas on which they plan to concentrate: desktop communications, desktop publishing, and desktop engineering. Does this mean that Apple is now a marketing company, examining needs before producing products? The answer to this question is fuzzy. Did anyone think they needed a desktop publishing system until the Macintosh and LaserWriter made it possible? Did Apple think about engineering applications of the Mac until engineers recognized the Mac's potential for CAD and bought lots of them? [In *On the Wing*, the Smithsonian Institution's film about Paul MacCready's project to build a flying model of a pterosaur, we see a Macintosh being used to simulate the motions of a flying dinosaur.] On the other hand, Apple clearly recognized the need for a simple, low-cost LAN before creating AppleTalk.

Being "market-driven" will not guarantee a business success; though listening to customers' expressions of needs is important, merely listening is not enough. Companies must be innovative, flexible, have the ability to both recognize opportunities *and* create new ones, *and* listen to their customers. Think of the history of the videocassette recorder. When SONY introduced the BetaMax, people thought the company was foolish. Most analysts thought no one would need or want to record

television programs. SONY was right about the VCR, but they were wrong about what features were most important to consumers. Consumers wanted long recording time; JVC, the developers of the VHS system, saw this, and VHS recorders could record 2 or even 4 hours of programming on a tape long before SONY incorporated similar capabilities in their machines. As a result of SONY's blunder, VHS machines dominate the market.

Still it would be foolish to call SONY a failure because they only control 15% of the VCR market. Fifteen percent of a big market is a lot of money. It's also dumb to say Apple is a failure because IBM sells more personal computers. Last time I heard, Apple had sold around a million Macs

---

When SONY introduced the BetaMax, people thought the company was foolish. Most analysts thought no one would need or want to record television programs.

---

and who knows how many Apple IIs. How can a $2 Billion company be a failure?

Further, I'm not sure it's accurate to say Apple hasn't been a marketing company. After all, Steve Wozniak designed the Apple I circuitry to suit himself and the other hackers to whom he hoped to sell computers. He got feedback from members of the Homebrew Computer Club in San Francisco. That sounds like marketing. Later, Steve Jobs decided the Apple II needed a friendly-looking simple plastic case and a colorful logo, so customers wouldn't be intimidated. That sounds like marketing. Apple decided that business users needed a computer designed for business, so they came up with the Apple III, which wasn't really a bad computer even though it eventually sold only 80,000 units. Marketing. In order to expand the market, they felt that computers had to

be easier to use. The result: the Lisa and Macintosh. Again, marketing.

Admittedly, not all of Apple's marketing decisions have been correct, but, then, IBM hasn't always been right either. Remember the PCjr and the PC portable?

How will Apple do in the future? The Macintosh Plus is a big improvement over the original 128K Mac; the Plus is also evidence that Apple is listening to the market, and customers have apparently responded. Macintosh sales are way up. I think Apple's future is bright, even if they don't have an MS-DOS computer.

More on Support

I knocked several computer stores last issue. It is with a feeling of justice that I report that one of them, the sorry Pasadena Computerland, has gone out of business.

I'd also like to dish out some praise. Scot Van Alstine is the Macintosh specialist at the Southpoint ComputerCraft store. The other day, I dropped by ComputerCraft to upgrade my Macintosh Pascal disks to version 2.0. Scot was busy giving personal instruction to a customer who was an obvious novice. Scot is generally helpful and enthusiastic, but this is the first time I had ever seen a knowledgeable computer store salesmen actually giving private lessons. I was impressed. When I asked for the *MacPascal* upgrade, Scot handed me the disks, pointed me to a Macintosh, and let me copy them myself. I haven't attended any meetings, but Scot has started a Macintosh users group which meets at the store once a month. I also think (I'm not positive) that Mac Pluses don't have those stupid plastic ComputerCraft logos stuck on the bezel anymore.

— *CLELAND EARLY*

Ponderings by Pete

This month I'd like to talk about the Macintosh. There is a lot going on in the Mac community, not all of it good. Since everyone is so sick of Steve Jobs, I'll refrain from commenting on that subject, except to say that Jobs' departure, ironically, might be the best thing that could have happened to the Apple //. Apple's realignment will take the // out of the back seat at Apple and put it back up front with the Mac, where it belongs.

On License Agreements

I am dismayed over the restrictive license agreements on new software packages. Microsoft and most others now give the user the "nonexclusive right" to use a package, usually with the limitation that the program may be used on one computer only. Presumably if you have one computer at home and one at work, you have to buy two copies of the same program. Also, if you sell your computer, you often cannot legally sell "your" program with it. At least Microsoft attempts to provide customer support and provides a form so you can report problems with the software.

Now, I can recognize the need for a license agreement, and it is possible that sales are increased by requiring one purchase per machine instead of one per user, but a while back I read a license agreement that just goes too far. My wife was helping to translate Palantir Software's *MacType* manual into French, and she asked me to boot up the program so she could see the program in action so she would better translate it. I took the time to read the license agreement and I was floored. I would like to share some excerpts with you (emphases are mine):

Opening this package can have the same effect on you as signing the software License Agreement card... If you are unwilling to sign the Agreement, return the software

UNOPENED...

I understand the software will **always remain your property:** even though I have paid a one-time fee for the **limited right to use it for up to 20 years,** if I break any of my promises below, you may choose to revoke that right...

I promise to keep the software **confidential...**

You promise that the software will do what you have said it **generally will** do, but you **do not promise:** [that] it will meet my particular needs; **[that] it will be trouble-free;** or [that] you will fix every problem.

I agree you can make changes in the software without letting me know and without giving them to me. If I want the changes **I may purchase them from you at your published price.**

I do not know what they mean by "limited right to use," nor do I know if Palantir will be dropping by in 20 years to pick up the disk (is the user required to keep Palantir informed of any change of address?), nor do I know how to keep the software "confidential." Is the software so bad they don't want me to tell anyone about it? Fortunately, I was not the one who opened the package, so hopefully I cannot be sued for not keeping it confidential. I cannot understand why software publishers include such insulting clauses in the license agreement, especially when they are so obviously unenforceable.

Now any competent publisher makes sure something is ready before releasing it. A book publisher does not publish a manuscript with proofreader's notes in the margin. Yet Palantir has deliberately released *MacType* before it was finished. To my amazement, when

I booted the disk and opened the notepad, I read, in part: "Need to deal with end-of-line character matching problems." The program is obviously not completely debugged, yet Palantir has gone out of its way to tell us that when the problem is fixed, they will not tell us nor will they provide us with the update. We will have to find out on our own (how, I don't know, since Palantir will not tell us and users have to keep it confidential) and buy it for an additional $49.95.

I do not intend this to be a review of *MacType*, which I have not even used, nor did I mean to pick on Palantir Software exclusively, for this type of problem is widespread today, especially with Macintosh software. I simply want to make people aware that today it is not enough to find out the technical aspects of a program, such as price, availability, and functionality. Legalities must be considered, and you must be careful to read between the lines. ***support shareware!***

I urge everyone with a Mac, or anyone planning to buy one, to avoid Palantir Software and any other company that tries to make us feel like criminals just for wanting to use the software we paid for. Restrictive policies against copying and the like are one thing, but this is absurd. For that matter, as a hard disk user, I'd like to go on record by urging everyone to make copy-protection one more criterion to use when evaluating software. Some software cannot be copied onto a hard disk, which is very inconvenient.

Copyrighted Icons?

Have you heard the one about copyrighted trash cans®? It seems Digital Research (DRI) was going to publish a program called *GEM Desktop* which was to be used on the Atari ST, among others. Apple blocked the release, however, claiming the program infringed on their Macintosh copyrights

by looking too much like the Macintosh desktop. Remember when Apple was stressing that the Mac interface was going to become the industry standard, and how wonderful they were for leading the computer world out of the wilderness? Well, now that people are starting to follow them, Apple starts acting like a corporation! What nerve!

Unfortunately, Apple may have bitten off more than they can chew. DRI is a fairly small company which Apple could intimidate, but the *GEM* decision will also affect the giants Atari and Texas Instruments, who will not back down. There will be further developments in this story. Meanwhile, back at Apple's corporate offices...

Hitting Below the Belt

Remember back when *MacBASIC* was going to set a new standard? When Apple spent hundreds of thousands of dollars developing the language? When Apple encouraged developers to write and sell books about the language? When people actually **bought** those books? Well, *MacBASIC* is no more. After negotiating with Microsoft, supplier of *Applesoft BASIC*, Apple decided to drop *MacBASIC*. Way back in the olden days, Apple paid $20,000 for the *Applesoft* license. That license was due to expire in 1986, and Microsoft was not going to renew it unless Apple dropped *MacBASIC*. Rather than spend hundreds of thousands of dollars to develop their own *Applesoft*, Apple agreed. So all of you who bought books about *MacBASIC* and all of you who waited for the release of *MacBASIC*, are out of luck. The reason for all this? It seems Microsoft was piqued because Apple's *BASIC* was better than theirs.

I think it is ridiculous that the Mac has no programming language bundled with it. Now that Apple's *MacBASIC* is dead (unless some enterprising hacker posts it on a bulletin board somewhere, hint, hint), it is even more important to find a good language. Assuming most users (as opposed to professional programmers) will be using BASIC, we are left with three major choices: *Microsoft BASIC* (Microsoft Corp, Bellevue, WA), *True BASIC* (True BASIC, Inc, Hanover, NH), and *ZBASIC* (ZEDCOR, Tucson, AZ).

Microsoft BASIC is the most expensive, least powerful, and most popular. Why, I don't know. Dealer base, I guess. When is a major magazine (who can afford to buy and test all three) going to do an impartial comparison? *True BASIC* seems to be the best, but also the most expensive (if you want to create clickable programs). *ZBASIC* seems to be the best value. It has a number of features the others don't (up to 54 digit accuracy, single step debugging), and automatically creates clickable applications without need for a $500 runtime package. It is fast and portable (same commands work on all computers). Seems like the best bet.

By the way, for you Pascal converts, there is a program called *P-Tral* ($125 from Woodchuck Industries, New York, 212-924-0576) which actually **translates** *Applesoft BASIC*

The Mac Plus is a true sales leader, moving at a clip of about 30,000 per month...

---

programs into Pascal source code. It is a // program, but should be of interest to you Mac programmers who use Meacom's *Mac+II* to emulate a //.

Who says you can't do multitasking on the Mac? Even Hertzfeld's *Servant* is not **true** multitasking. It just looks like the Lisa interface (anyone old enough to remember that?) with several applications open at once in different windows. However, *MachI* from The Palo Alto Shipping Co. (415-854-7994) promises **full multitasking** in a Forth environment. The cost is $50. It supports MacInTalk, Appletalk, and creates standalone programs with no licensing fees.

Last, for (all (of you (LISP (fanatics)))), there is *MacScheme*, which may supplant ExperTelligence's buggy *ExperLISP*. *MacScheme* is only $125, compared to $495 for *ExperLISP*.

Mac Plus Department

Macpack (Seattle, WA, 1-800-228-7042) makes a Mac Plus compatible carrying case. It holds the Mac Plus

keyboard **and** a HD20. The price is $65.

If you need more "circular-8" connectors, contact C Enterprises at 1-619-744-8182. Note that "Apple does not endorse this company." Probably because the price is lower. By the way, what is the point of these silly little things? They provide no power and they tend to **fall out!**

The Mac Plus is a true sales leader, moving at a clip of about 30,000 per month (compared to 10,000 for the 512 before the Plus). Bolstered by Mac Plus sales and the previous volume leader, the //c, Apple led all micro manufacturers in sales again in February, although IBM led in dollar volume. The Imagewriter II was the single best-selling printer in March; the Epson line was the best-selling line. Apple, led by the LaserWriter, was the dollar volume leader among printers. In software, *Excel* was the best selling Macintosh program, followed by Microsoft *Word* and Aldus *Pagemaker*. All in all, Microsoft had **over 53%** of the dollar volume in Mac software!

The Rumor Mill

Microsoft is about to announce a BASIC **compiler** for the Mac. Prototypes of a 640 x 480 pixel **color** Mac (code-named "Jonathon") are floating around Cupertino, although one of the engineers reportedly left Apple over the pricing policy. Apple has set the prerelease price at $2495. Apple is also working on a 68020-based (read: awesome) Mac and an **expandable** Mac, the latter likely to be ready by October for the consortium universities. It should have a larger screen and optional MS-DOS.

The **new** Apple //x is going to be based on the 65SC816 processor which will emulate the Mac, the //e and the //c (this chip emulates a 6502 and a 68000). Its optional 3.5" drives might read both // disks and Mac disks. This would be in keeping with Apple's current policy of uniting the two product lines. Current information from Apple states that the machine will be ready by September and might have a "basic system" price of as low as $1400. The machine will have 512K of RAM (expandable to 4.25 Meg), 768x512 pixel, 128 color graphics

(from a palette of 4096), serial port, printer port, SCSI port, 5-7 expansion slots (probably compatible with //e slots), and a built-in mouse and disk controller. And yes, it has a detachable keyboard! Best of all, Apple is working on a cross-development system to allow Mac programs to be ported to the //x.

Colorocs Corporation of Norcross, GA expects to release a very fast, full-color, laser page printer in 1987. Unfortunately, the price is expected to be about $20,000. But who knows? Industry experts predict that page printer prices will plummet to the $1,500 range by late 1986. Apple, by the way, has dropped all its list prices by 20-35% recently. And Texas Instruments' Omni series of laser printers should force Apple's LaserWriter prices down by **another** $1000-2000. By the way, look for a LaserWriter designed specifically for the //c soon. (What about the //e? Is Apple abandoning it? Pretty doubtful. Anyway, you can already use a LaserWriter on the //e — just ask Don Lancaster!)

Questions and Answers Department

Bill Hailey: Thanks for all the tips! In answer to your question in the Aug/Sept *Apple Barrel*, you clear a page break in *MacWrite* by clicking after the last character **before** the break and dragging over the break to select it, then typing <Backspace>. Here's one for all you *MacTerminal* users: To avoid the "Are you sure..." dialog box when resetting or clearing lines off top, hold the **option** key down when selecting these menu items.

Everyone: buy a keypad if you use a spreadsheet. You won't believe the difference that $90 makes! (Why is it so few companies outside Microsoft take advantage of the **cursor keys?!?**) I also have a MousePad ($10) from Moustrak, Inc, and I haven't had to clean my mouse **once** since I got it! Fortunately, my company pays for all this.

Anyone: I want an Apple //e speedup card. Can anyone recommend the best one? From what I've heard, that would be TransWarp, but I'm concerned about price. Also, if it only has 256K of RAM, how does it work with, say, a 512K Ramworks?

Rhetorical Questions Department

Everyone says the Mac is fast, it's the software that's slow. Surely Apple would agree with this. So tell me, why does the *Finder* waste time redrawing a window when it's just going to erase it anyway? For instance, when I choose Quit, the area under the File menu is redrawn before quitting. This takes several seconds. Why bother? Also, in *Macwrite 4.5* and Microsoft *Word*, if you open a *Macwrite 2.2* file, it says, "The document is being converted,"

Best of all...a cross-development system to allow Mac programs to be ported to the //x.

with an OK button. Why is there no Cancel button?

One Last Tip Before I Sign Off Department

By now everyone has heard how great the *Resource Editor* is for adding keyboard shortcuts, customizing icons, etc. I use it for one more thing, though. By accessing dialog boxes, I can copy the ✓, ⌘, and characters onto the notepad for ready reference in *MacWrite*. That way, when I mean ⌘-P to print, I can say ⌘-P to print! All have to do is copy them from the notepad, paste them into my document, and change them to Chicago-12. (The LaserWriter will not print characters with an ASCII value less than 32. If you plan to use this tip to print on the Imagewriter, fine, but to access these characters on the LaserWriter, you will need to use *ResEdit* to copy them from ASCII 17, 18, and 20 in Chicago-12 to three unassigned option characters. You can then find the correct key sequence to type them with *Key Caps*. I have put a patched Chicago-12 in the HAAUG library. Be sure not to use font substitution when printing!)

— *PETER LEMETTAIS*

Empirical Fumblings and Arcane Language
(Adoption suggestions for a CPU)

Empirical means depending on experience or observation without due regard to science and theory. Fumbling is to feel or grope about clumsily. That's where most of us start, deriving comfort only from two facts. There is no keyboard command that should be able to hurt a properly designed microcomputer. It requires a carefully written program to do damage. Also, the devices come with a warranty. You have a right to try things and see what happens. Exceptions are hand grenades, pry-bars, and such.

Arcane infers secret or mysterious. DP (Data Processing) managers, MIS (Management Information Systems) personnel, consultants, and programmers work hard at undecipherable communications. We have learned that the less somebody understands about what we're doing, the more we can charge for doing it. We're simply following a pattern long established by doctors, attorneys, and automobile service departments. Picture calling a consultant and asking to have an immediate meeting. A reply such as "Is it important that we meet now" is $20 an hour level. A more thoughtful "Is it imperative that we interface at this juncture in time" indicates $100 an hour talent.

It's very hard to get rid of a person for not doing his job if you can't figure out what his job is. Incomprehensible terms are a professional requirement.

The CPU is the Central Processing Unit, the brain of the package, quite a separate entity from keyboards, disk drives, monitors, and cables. In the beginning it's the most expensive part of the package. As one adds the peripheral equipment and software, it becomes relatively trivial, somewhat like a marriage license fee. Since it does start out as the major budget item in the purchase, let us start out by giving consideration to what to buy and how best to protect our investment.

First recommendation

Talk it over with users. Not just the CPU but the entire proposed package. People have left stores with printers that are unsupported by the machine they purchased at the same time. Unsupported meaning that the printer can be made to work properly with the machine but it might take a couple of months by an assembly language programmer working at a systems interface level to make it happen. And of course there are many items that may be excellent, but unsuited for what you have in mind. There are only about three stores in Houston that seem to me to be both knowledgeable and helpful. They too must sell what they stock if they are to survive.

Start with your problem. What do you want done by your computer. This

There are only two classes of computers — experimental and obsolete — and only one class you can get delivery on.

leads naturally to the selection of software and lastly to the selection of a computer.

Start with your problem. What do you want done by your computer. This leads naturally to the selection of software and lastly to the selection of a computer.

The above paragraph was repeated deliberately. It is impossible to overstate the importance of this simple progression, identifying what you want to do first, then the software. The last item to decide on is the computer.

There are a lot of experienced, professional users of every conceivable type of equipment and combination at

computer clubs. Some (like me) love to be helpful because it's a modest way to brag and show off what we know. While conversation at a club is fun, talk is not cheap. Many of us are well paid by industry, for exactly the same advice, during business hours.

Second recommendation

If you haven't already purchased a microcomputer, don't buy one until you're ready to plug it in and start using it as soon as you can transport it to the intended location. Prices keep dropping, new products keep appearing, old ones keep getting improved. Computers and their little brothers, the electronic calculators, are one field that has delivered more for the dollar every year since the first commercial model hit the market. And the buyers situation is still improving at an exponential rate. When you are ready to use it, go for it. A futile reason to wait is that prices are dropping or new models are coming out. That's been true since day one of the industry, and will continue true until everyone able to read this article has expired of old age, including direct genetic descendents of Methusela. There are only two classes of computers — experimental and obsolete — and only one class you can get delivery on.

The areas where I would let a salesman talk upscale to me are memory and disk storage capacity. To serious users there is no such thing as enough of either.

Next recommendation

Actually make a pass at trying to read the manuals on setting up your system, before you do it. They may give you some useful information. On an Apple //e, one quite naturally plunks the monitor on top of the CPU and all is well. Placed on a IIc most monitors have enough stray electromagnetic fields to cause disk read errors. They block

Baxt & Associates
Houston's First Laser
Phototypesetting Service Bureau

Your Pagemaker documents created on your Apple Macintosh can now be output in Houston for significant savings in time and money on the laser typesetter of the highest resolution available (2500 d.p.i.).

Using the PostScript page description language, you can create fully paginated pages. Eliminate double keying, repetitious proofing, and complicated paste-up. Gain control over page layout.

Call 661-7933 for your free consultation on cost-cutting measures.

Other services include design, typography, graphics, and copywriting

Baxt & Associates   4828 Loop Central, #820, Houston, TX 77081

Apple Barrel   Page 13

Stickybear Blues
and Reds and Greens...

Okay, I admit it. After I've spent time reconfiguring hard disk systems on other people's (dare I say it?) IBM Compatibles, I like to come home and play with *Print Shop* on my Apple //e. For my kids, of course. We'll pick out the graphics and font styles for a greeting card, and they choose what to say. After we print it out, they really enjoy coloring it. This way, all their friends (and mine) have cards sent to them with their own name on it and a suitable message. Large banners for graduation, promotions, births, etc. can be created and colored in too. This has afforded us great entertainment together (and sometimes a quiet moment for Mom while the kids emblazoned their masterpieces). But something was missing. Printing with only black ink (*Print Shop* does not support color printers) on an Imagewriter II Color Printer is actually a physically painful experience.

Then I see an ad announcing *Stickybear Printer*, with Color Capabilities. My children and I are well-versed in Stickybear software. My three year old daughter learned her ABCs and Numbers using their amusing graphic-and ditty programs (of course, she thinks that "eightoctopuses" is one word). My six year old son was first introduced to arcade-type hand and eye coordination practice with *Stickybear Basket Bounce* and *Stickybear Bop* ("so-so" software—"so" the children are coordinated enough to use the program, "so" they will soon be bored with it). They both have enjoyed the *Opposites* and the *Shapes* programs (the ones I rate the highest). Therefore if Color was to come to my Imagewriter II via *Stickybear*, then we had Great Expectations!

Well, I purchase my long-awaited *Stickybear Printer* software at the HAAUG meeting (where else?), drive straight home, and load it on my 320K (thank you RamWorks) Apple IIe.

Boy, is this program slow! The program is menu-oriented just like *Print Shop*, and moving around the choices on the screen makes me think it should be called Stickytortoise. But unlike *Print Shop* on an Apple, *Stickybear Printer* displays the graphics on the screen as you choose them (the IBM version of *Print Shop* will display your graphics too, but only after you are finished creating). You can even move the graphics around the screen for better positioning. After I select a Background-Bricks, Snow Scene, Space Scene, etc., I load Picture after Picture of Dog, Cat,

---

Therefore if color was to come to my ImageWriter II via Stickybear, then we had Great Expectations!

Horse, Tree, Flower, and several styles of Stickybears (the latter which seem to be the least sharp graphics images). I notice the color from these graphics seem to bleed onto the Background colors on my monitor (why oh why didn't I go the extra dollars for the RGB high resolution instead of the Color Monitor?). Well, this turns out to be a problem not only with the monitor, because it later prints out like that too. "What-you-see-is-what-you-get" software sometimes has its drawbacks.

Okay. Let's write some Text. There are a few very plain Fonts of different sizes, available in Black or White. On a Greeting Card, the only font size that seems to fit well is the Small Face Black or White, and even then, the word "Grandmother" has to be hyphenated in the middle and written on two lines.

Now we have created both the Outside and the Inside of the card. We even Store what we have designed on a diskette (the Apple version of *Print Shop* doesn't offer this feature, but the IBM version does). So let's use that color printer! (Make sure to install the color ribbon first.) Does it look like the slick advertising you see in the computer magazines? No. The kids are terribly disappointed after the twenty minutes it takes to play and make all our choices. My son's only comment is "I can color better than that."

Ever optimistic, I say "That's okay, guys. Wait till you see how you can design a 3-D Cutout House!" After 15 minutes more of picking what Stickybears to apply to what sides of the house, we're ready to print again! When we cut it out and oh-so-carefully tape it together, it measures about 1 1/2 inches by 2 inches by 1 1/2 inches. "Mom, can we make a bigger one next time?" "No." "Doesn't it make a two-story?" "No." "Mom, can we go clean up our rooms, or something?" "Oh, alright." Now they let me play with it anytime I want, with no interruptions.

I would have liked to see a little more polished version of *Stickybear Printer*, with no colors running, a little sharper graphics quality (especially the bears), and a little larger cutout house or two (they did have a great idea there). Nontheless, the *Stickybear* Series in general provides my children with a friendly introduction to the computer world and all its wonderfully abstract technology. And so what if they do a little learning on the side? Just color me Stickybear.

—GINNY MORLEY

Stickybear Printer
   Weekly Reader Family Software
   Developed by Optimum Resource, Inc.
Requires: an Apple II, II+, IIe, or IIc with at least 48k memory and one disk drive, and supports the Apple Scribe, black or color ribbon, Imagewriter II, black or color ribbon, C. Itoh Prowriter, Epson FX 80, and Okidata 82, 92.

AppleWorks Spreadsheet Tips

The ability to easily copy text, values, and formulas to other parts of the same spreadsheet or to other spreadsheets is one of the most powerful features of the *AppleWorks* spreadsheet program. The sophisticated use of Open-Apple C (Copy) greatly reduces the time required to create a spreadsheet and to transfer data between spreadsheet files.

After pressing Open-Apple C the user is offered the choice of staying within the spreadsheet or of using the Clipboard. Although one normally thinks of using the Clipboard to transfer information between files, the Clipboard can also be used to copy blocks of data within the source spreadsheet. For example if you have set up several rows of labels, ie.,

```
Jan    Feb    Mar    etc
1986   1986   1986
==========================
```

and wish to copy this block of data several times so that each printed page has a page header, it is best to use the Clipboard. Use Open-Apple C To Clipboard and "paint" the entire block then put the cursor where you want it and press Open-Apple C From Clipboard. The Clipboard will remember its last batch of data so there is no need to "repaint" if you want to repeat the copy several times. Be aware when using the Clipboard that you must copy an entire row (A to DW), not just part of a row, and you can not copy a column.

Most of the time you will copy within the worksheet and not use the Clipboard feature. Now you are limited to copying the cells on either a single line or a single column. It is not yet possible to copy a block having multiple lines and columns eg., a five row by ten column block. The tip here is to copy by row if you have fewer rows than columns or copy by column if you have fewer columns than rows. It also may be easier to copy one or more cells

you would not normally copy and then change the unwanted cells later if it allows you to copy a larger block. For example, to copy A1 through A10 and then A12 through A25 consider copying A1 through A25 and correcting the entry at A11 after you have made the copy.

If you copy within a worksheet to an area already containing entries, the copy will replace the entries. If you use the Clipboard this does not happen. Instead the entry is displaced the distance necessary to insert the copy by automatically inserting the rows needed.

AppleWorks will not allow you to copy to an area that is part of your source entry ie., you can't copy A1 through A10 to A5 through A15 as there is an overlap from A5 to A10. To get around this copy to a blank area of the spreadsheet then copy back to the location that you want it.

Copying formulas is where the electronic spreadsheet really saves work. The user can set up all of the formulas pertaining to a months activities in a column and then in almost no time copy that column across the spreadsheet for the entire year. TIP-It is far easier to do this if you have not yet entered any values, otherwise the values will get copied too.

After copying a formula within the worksheet the computer will highlight each term and ask "No Change or Relative." Usually the answer is "Relative" which means that the cell highlighted will be automatically changed to the cell in the same relative position at the new location. If, however, you do want to keep the highlighted quantity unchanged then respond "No Change." When using the Clipboard any formulas that have been copied will automatically be "Relative."

Be careful when indicating "No Change" or "Relative", as a slip of the finger when working with many formulas can give bad results which

may puzzle you for hours. If this happens, review each formula; the chances are likely you hit "No Change" when you meant to hit "Relative."

If you want to copy spreadsheet data to the word processor you do not use Open-Apple C to copy to the Clipboard. The data must be printed to the Clipboard using Open-Apple P (Print). Be sure that the line length of the spreadsheet data transferred is less than the current line length in the word processor or you will get a wraparound. To eliminate this wraparound, either change the margins in the word processor or transfer shorter lines. If you must transfer a long line, then in the word processor change to compressed print for the entry. The screen will still show wrap around but your printed copy will come out correctly.

Probably the most sophisticated use of the copy feature is in the creation and use of spreadsheet templates. If you create a spreadsheet that you know you will use again in the future but with different data eg, next year's budget etc, work up the spreadsheet format and then copy it to a separate file by using the Clipboard before you start entering data. When you need it again pull it back on to the desktop and enter the new numbers. The use of templates greatly simplifies your work and they can be easily revised if the occasion arises. Templates are available commercially where all of the setup and technical programming have been done for you.
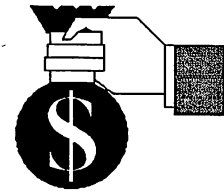
Next month this column will cover the use of formulas and continue discussing the Open-Apple features.

Final tip — Want to hide a column when you print your spreadsheet? Before printing use Open-Apple L and reduce the column width to 1 character. All numbers in the column become #, yet the formulas still work!

— *JOHN B. SLACK*
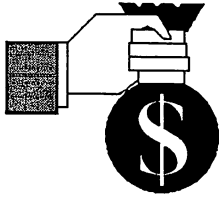
HERE IT IS!!
TAX PRO 86

Don't be surprised on April 15th 1987!! By using the TAXPRO 86 spreadsheet template with AppleWorks you can determine on a daily basis your tax situation and the amount of money to be owed to or refunded by the IRS when your 1986 tax return is filed.

TAX PRO 86 is a spreadsheet template designed for use with AppleWorks which permits the user to keep track of his current tax status throughout the year. It is for the average taxpayer who has multiple sources of income ie.,wages, spouse wages, interest, dividends etc and who itemizes his deductions. The tax tables and calculations contained in TAX PRO 86 are those currently in effect for 1986. TAX PRO 86, TAX PRO 87, TAX PRO 88, etc will be issued as the tax rates and tax code change in later years.

TAX PRO 86 may be enhanced by the user as required to accomodate those transactions that occur which are not normally encountered by the average taxpayer. The intent of TAX PRO 86 is to simplify your tax situation in an easy to use manner so that you are not surprised on April 15th!!

Like FINANCIAL PRO, TAX PRO 86 is easy to use. Once the financial data is entered, the results are immediately available. By using the powerful AppleWorks spreadsheet, "What If?" scenarios can be presented for evaluation.

Since TAX PRO 86 generally follows the format of the IRS tax forms, the preparation of your tax return is made much easier.

For a limited time HAAUG members can buy TAX PRO 86 for only $20.00. To order your copy of TAX PRO 86 call 713-491-1747 after 6:00 PM or send a check or money order with the following coupon.

Pro Products
1179 Bournewood
Sugar Land, TX 77478
Please send me _____ copies of TAX PRO 86 @ $20.00 each.

NAME_____

ADDRESS_____

CITY_____STATE_____ZIP_____

The ProDOS SET.FILE.TYPE Routine
A Telecommunications Utility

There is no standard for transferring file type data (BIN, BAS, TXT, etc) during Bulletin Board downloads in the Apple][world (*Ascii Express* has developed a format, but it is not yet universally accepted). Therefore, communications programs generally don't know what type file they are receiving during an XMODEM transfer and they must use some default type when storing the file. For instance, some use a "Typeless" file ($00) while others use a TXT file, etc.

As a result, most bulletin boards end up providing their program downloads in the form of text files which must be exec'ed or otherwise processed to return the file to its original form.

This technique works, but has drawbacks. First, a program stored in a text file uses extra BBS storage space and significantly lengthens download time since it is two to three times larger than the original "true" file. Second, the exec process is subject to errors if the file's creator wasn't careful to set up all the proper parameters so that the text file would exec and save to disk properly.

One of ProDOS' advantages over DOS 3.3 is the flexibility it provides in handling its wide range of file types. For instance, you can BLOAD and BSAVE any file, no matter what type it is: Text, Basic, System, *AppleWorks* Data Base, etc.

A file's attributes can also be changed so a BIN file can be given a new loading address, it can be changed from a TXT to a BAS file, etc. without exec'ing it. Note that the actual data in the file is not changed, only how the file is identified and therefore how ProDOS handles it.

The SET.FILE.TYPE routine was written to take advantage of this flexibility and potentially simplify telecommunications file transfers. If your communications software has XMODEM (or other supported error checking) capability then use it to download the true native state file from the bulletin board if it is available rather than the text file version.

Once you have the file stored on disk, exit to Basic, BRUN SET.FILE.TYPE and change the file from your software's default type to its original type (the Starting Address is also necessary for BIN files). The original file type and other needed data should be available in the file description on the bulletin board. The file will then be "ready to run." No muss, no fuss, no exec'ing.

The operation of SET.FILE.TYPE is self explanatory. Simply enter a file name with or without a prefix (or a "?" for a catalog). The current file type will be displayed and you will be asked for the new type three letter designation. A list is supplied on screen. You will

```
1    ********************************
2    *                              *
3    *    CHANGE FILE TYPE ROUTINE   *
4    *           E. L. Oshlo         *
5    *                              *
6    ********************************
7
8    HOME      EQU    $FC58
9    CLREOP    EQU    $FC42
10   COUT      EQU    $FDED
11   CROUT     EQU    $FD8E
12   BUF       EQU    $200
13   FN        EQU    $280       Pathname storage spot
14   GTLEN1    EQU    $FD6F
15   KEYIN     EQU    $FD1B
16   CURSUP    EQU    $FC1A      move cursor up 1 line
17   WINTOP    EQU    $22        top of text window
18   CH        EQU    $24
19   DOSCMD    EQU    $BE03      Basic.System Dos Command call
20   COL       EQU    $C01F      80 column active indicator
21   COLF      EQU    $6         My 80 column flag
22   MLI       EQU    $BF00
23   PRTERR    EQU    $BE0C
24   BADCALL   EQU    $BE8B
25   PRBL2     EQU    $F94A      Print Blanks
26   ZMODE     EQU    $FFC7
27   GETNUM    EQU    $FFA7
28   A2        EQU    $3E
29   FC        EQU    24         Supported file count * 3
30
31   *
32   * INSTRUCTIONS FOR ADDING ADDITIONAL FILE TYPES:
33   * ------------------------------------------------
34   * Store # of supported files times 3 in FC equate above.
35   * Store the following data in the same relative position
36   *    in each of the 3 arrays at the end of the source:
37   * 1. ProDos file type # followed by 0000 in FTNO array
38   * 2. ProDos 3 letter designator in FT array.  (Leave
39   *    $00 last in array and don't count it for FC calc.)
40   * 3. File type  default auxtype (load address) followed
41   *    followed by 00 AUXTP array
42   *
43         ORG    $1000
44
45   * Remove JSR $C300 to test for 40 or 80 column display
46
47         JSR    $C300
48         LDA    #3          Initialize CAT/CATALOG routine
49         STA    FCAT+1
```

also be prompted for the Starting Address in Hexdecimal form for new BIN files. Enter a "Q" at any data input point to quit. The routine runs in either 80 or 40 column mode depending on which you were in when starting the routine. The only noticeable difference is the form of the catalog provided when you enter "?."

SET.FILE.TYPE is initially set up to handle seven file types: TXT (text, sequential only), BIN (binary), SYS (system), ADB (*AppleWorks* Data Base), AWP (*AppleWorks* Word Processor), and ASP (*AppleWorks* Spreadsheet). In addition, it will accept a $00 (typeless) file as a current file type but will not let you change a file to a typeless one.

DIR files and some other types are not supported to help prevent mistakes. Random access text files are not supported since I couldn't think of a reason to convert a random access data file to another type.

The SET.FILE.TYPE source code can be modified easily to support additional file types. See the notes in the Merlin Pro assembler source code in the listing.

SET.FILE.TYPE can also be used with DOS 3.3 files if they are first converted to ProDOS. After the file type has been changed, convert it back to DOS 3.3. Note that DOS 3.3 only supports three of the file types: BIN, BAS, and TXT.

If the bulletin board doesn't provide "true" files, talk to the SYSOP through the board's feedback function. The Houston Area Apple User's Group HAAUG Heaven board does provide true files for downloading.

SET.FILE.TYPE is available for downloading (as a text file that must be exec'ed!) on the HAAUG Heaven board in the Apple][Utilities section. Hopefully it is the last file from HAAUG Heaven you will have to exec. The Merlin.Pro assembler source code is provided below and on HAAUG Heaven for those with an interest in programming in assembly language. SET.FILE.TYPE uses the ProDOS Machine Language Interface to make Prefix, Get.File.Info, and Set.File.Info calls. They share a common parameter list at the end of the routine. Only part of the Set/Get File Info list was

```
50            LDA   #8          Initialize # spaces to Print
51            STA   COLF
52            LDA   COL         Test for 80 col card active
53            BPL   TEXT        > 127 yes, <128, no
54            LDA   #28         Print 28 spaces if active
55            STA   COLF
56            LDA   #7          Use CATALOG command if active
57            STA   FCAT+1
58    TEXT    LDA   #00
59            STA   WINTOP      TEXT
60            JSR   HOME
61            LDX   COLF
62            JSR   PRBL2       Print COLF spaces
63            LDX   #00
64    :LOOP   LDA   TITLE,X     Print & underline title
65            JSR   COUT
66            INX
67            CPX   #25
68            BNE   :LOOP
69            LDX   COLF
70            JSR   PRBL2
71            LDX   #24
72            LDA   #"-"
73    :LOOP1  JSR   COUT
74            DEX
75            BNE   :LOOP1
76    START   LDA   #3
77            STA   WINTOP
78            JSR   HOME
79            LDX   #22
80    :LOOP2  LDA   QPMT,X
81            JSR   COUT
82            DEX
83            BPL   :LOOP2
84            LDA   #5
85            STA   WINTOP
86            JSR   HOME
87            LDX   #14
88    :LOOP3  LDA   FNPMT,X     Print File Pathname Prompt
89            JSR   COUT
90            DEX
91            BPL   :LOOP3
92            JSR   GTLEN1      Get Pathname
93            STX   FN          Store Pathname length
94    :LOOP4  LDA   BUF-1,X     Move Pathname to FN
95            STA   FN,X
96            DEX
97            BNE   :LOOP4
98            LDX   FN
99            LDA   BUF-1,X     Get last Pathname Character
100           CMP   #"?"        If it's a ?, then catalog
101           BEQ   NEXT1
102           AND   #%11011111  upper case conversion
103           CMP   #"Q"        If it's a Q then end
104           BNE   NEXT2
105           BEQ   END
106   NEXT1   DEC   FN          Decrease Pathname Length by 1
107           BEQ   CATA        to erase "?"
108           LDA   #1
109           STA   MLIPL
110           JSR   MLI         Set prefix if new one was given
111           DB    $C6         with the ?
112           DW    MLIPL
113           BNE   ERROR
114   CATA    LDX   #00
115   LOOP2   LDA   CAT,X       Store CAT<CR> in input buffer
116           STA   BUF,X
117           INX
118   FCAT    CPX   #3
119           BCC   LOOP2
120           LDA   #$8D
121           STA   BUF,X
122           LDA   #7
123           STA   WINTOP
124           LDA   #00
125           STA   $BEOF
```

```
126          JSR    DOSCMD      Execute CAT/CATALOG command
127          LDA    $BE0F       Load Basic.system error code
128          BEQ    PRESPT      if no error, Press a key
129  ERROR   JSR    BADCALL
130          JSR    PRTERR
131  PRESPT  JSR    CROUT
132          LDX    #15
133  :LOOP   LDA    PAK,X       print press a key prompt
134          JSR    COUT
135          DEX
136          BPL    :LOOP
137          JSR    KEYIN       wait for keypress
138          JMP    START       start over
139  END     JSR    CROUT
140          LDA    #00         text
141          STA    WINTOP
142          JMP    $BE00
143  NEXT2   JSR    CROUT
144          LDA    #$A         execute Get File Info MLI call
145          STA    MLIPL
146          JSR    MLI
147          DB     $C4
148          DW     MLIPL
149          BNE    ERROR
150          LDA    FTYPE       Get file type code & decipher
151          LDX    #00
152  :LOOP1  CMP    FTNO,X      using FTNO as the key
153          BEQ    PTFT
154          INX                look at every 3rd FTNO - 2 dummy
155          INX                bytes between each to make index
156          INX                match 3 char file type codes
157          CPX    #FC+3
158          BCC    :LOOP1
159          JSR    CROUT
160          JSR    CROUT
161          LDX    #FC
162  :LOOP2  LDA    NOSUP,X     if file type no. not found in
163          JSR    COUT        the list then print not
164          DEX                supported type message
165          BPL    :LOOP2
166          JMP    PRESPT      wait for key press
167  PTFT    LDY    #20
168  :LOOP   LDA    CFT,Y       Print Current type title
169          JSR    COUT
170          DEY
171          BPL    :LOOP
172          LDY    #00
173  :LOOP1  LDA    FT,X        Print file type
174          JSR    COUT
175          INX
176          INY
177          CPY    #$3
178          BNE    :LOOP1
179          JSR    CROUT
180  NFLOOP  LDX    #23
181  :LOOP   LDA    NFT,X       Print new file type prompt
182          JSR    COUT
183          DEX
184          BPL    :LOOP
185          JSR    CROUT
186          JSR    CROUT
187          LDA    #"("        print list of file types
188          JSR    COUT        with space between each
189          LDX    #00
190  :LOOP1  LDY    #03
191  :LOOP2  LDA    FT,X
192          JSR    COUT
193          INX
194          CPX    #FC
195          BCS    NF
196          DEY
197          BNE    :LOOP2
198          LDA    #" "
199          JSR    COUT
200          JMP    :LOOP1
201  NF      LDA    #")"
```

actually coded into the program. Parallel data tables at labels FTNO, FTYPE, and AUXTP are used to make the conversion between ProDOS' one byte type codes and the three letter file designations. Two dummy bytes were inserted between each FTNO data byte to make the table the same length as the FTYPE table and avoid having to multiply or divide the index by 3 when moving between them. As with most programs, most of the code deals with input/output to the screen. The actual file processing is a relatively small portion of the total file. Most of the major routines are outlined in the source file comments.

— RICK OSHLO

Changes have been made in the Set.File.Type program. The updated version is available on HAAUG Heaven.

— CLF

```
202            JSR    COUT
203            JSR    CURSUP    move cursor up 2 lines
204            JSR    CURSUP
205            LDA    #21
206            STA    CH        Move cursor to column 21
207            JSR    GTLEN1    Get new file type
208            LDA    BUF
209            AND    #%11011111 upper case conversion
210            CMP    #"Q"
211            BNE    :LOOP
212            JMP    END
213  :LOOP     CPX    #3
214            BEQ    NEXT8
215  REPEAT    JSR    CURSUP
216            JMP    NFLOOP
217  NEXT8     LDA    BUF-1,X    Set buffer to upper case
218            AND    #%11011111
219            STA    BUF-1,X
220            DEX
221            BPL    NEXT8
222            LDX    #00        Initialize X & Y for scan of
223            LDY    #00        ASCII file types.  Look for a
224  LOOPY     LDA    BUF,Y      match with Input BUF
225            CMP    FT,X
226            BEQ    INYR       if FT/BUF match inc Y register
227            CPY    #1         Start Y over if no match &
228            BCS    RESET      it's not 0
229            INX              Inc X for next char in FT
230            CPX    #FC
231            BCS    REPEAT     Back to prompt if no match
232            BCC    LOOPY      If match,get next chr w/same Y
233  INYR      INY
234            CPY    #3         Match 3 consecutive chars
235            BCS    CONTINUE   if so bail out of loop
236            INX
237            JMP    LOOPY
238  RESET     LDY    #00        Reset to 1st BUF character
239            JMP    LOOPY      Start search over
240  CONTINUE  DEX              Back X up to beginning of 3
241            DEX              character match in FT string
242            LDA    FTNO,X     Load MLI file type using 3
243            STA    FTYPE      char match in FT as a key
244            CMP    #6         BIN file?
245            BEQ    FADDR      If yes, get file load address
246            LDA    AUXTP+1,X
247            LDY    AUXTP,X
248            TAX
249            JMP    AUXSET
250  FADDR     JSR    CLREOP     clear to end of window
251            LDX    #20
252  :LOOP     LDA    LENPMT,X   Print Start address prompt
253            JSR    COUT
254            DEX
255            BPL    :LOOP
256            JSR    GTLEN1
257            LDA    BUF
258            AND    #%11011111 upper case conversion
259            CMP    #"Q"
260            BNE    :CONT
261            JMP    END
262  :CONT     CPX    #5         Reject if len > 4
263            BCC    CONVERT
264  CURS      JSR    CURSUP
265            JMP    FADDR
266  CONVERT   JSR    ZMODE      Convert ASCII to Hex data
267            JSR    GETNUM
268            LDX    A2
269            LDY    A2+1
270            CPY    #$95       Reject if >= $95FF
271            BCS    CURS
272            JSR    CROUT
273  AUXSET    TYA              store Y on stack since CLREOP
274            PHA              scrambles A & Y registers
275            JSR    CLREOP     clear type list from screen
276            PLA
277            STX    AUXTYPE
278            STA    AUXTYPE+1
279            LDA    #7
280            STA    MLIPL      MLI Set File Info call
281            JSR    MLI
282            DB     $C3
283            DW     MLIPL
284            BEQ    RESTART
285            JMP    ERROR
286  RESTART   JMP    PRESPT


289  TITLE     ASC    "CHANGE FILE TYPE ROUTINE",8D
290  FNPMT     REV    "File Pathname: "
291  LENPMT    REV    "Starting Address...:$"
292  CAT       ASC    "CATALOG"
293  CFT       REV    "Current File Type..: "
294  NFT       REV    "Enter New File Type: ___ "
295  FTNO      HEX    0400000600000700001900001A0000
                     1B0000FC0000FF000000
296  FT        ASC    "TXTBINFNTADBAWPASPBASSYS$00"
297  AUXTP     HEX    0000000000000000000000000000000
                     0000000801002000000000
298  PAK       REV    "  Press any Key "
299  NOSUP     REV    "Unsupported File Type"
300            DB     $87        bell
301  QPMT      REV    "(? = Catalog, Q = Quit)"
302
303  MLIPL     DB     $A         MLI Call Parm List
304            DW     FN
305            DB     $00        access (supplied by GFI call)
306  FTYPE     DB     $00        file type
307  AUXTYPE   DB     $00,$00    Lo/Hi start address
```

--End assembly, 809 bytes, Errors: 0

Apple //

Apple Access	Patrick Hennessey	342-0643
AppleWorks	Jack Cowart	467-4215
	John Slack	491-1747
AppleWriter //	Robin Cox	778-1635
AppleWriter ///	Shorty Hirsekron	697-9797
CAD-Apple	Dick Fairman	723-9619
dBase //	H. Helm	529-7405
Data Factory	Dick Fairman	723-9619
Dollars and Sense	Leon Cesazini	467-3650
DowJones Market		
Analyzer	Ed Hurley	497-8877
Home Accountant	Rick Oshlo	578-8530
IAC Calc	Chris Kravits	320-0381
Letter Perfect	Bill Muhlhausen	668-3963
List Handler	Rick Kravits	320-0381
MCI Mail	Patrick Hennessey	342-0643
Multiplan	Mackie Brown	688-8655
	Jack Bailie	440-5963
	Glen Ray	359-2103
MultiScribe	Henry Bernstein	728-0081
Palantir	Re Kramer	358-6687
PFS	Jack Cowart	467-4215
	Re Kramer	358-6687
	Ed Haymes	522-1863
Pie Writer	Ray Fielding	358-4217
Quickfile	Jack Cowart	467-4215
ScreenWriter //	Dick Fairman	723-9619
Space Eggs	T. Sloan Engle	774-9055
Time Is Money	Dick Fairman	723-9619
VersaForm ///	Shorty Hirsekron	697-9797
Wordstar	Ed Haymes	522-1863
Word Handler	Rick Kravits	320-0381

Would you like to be a specialist? Leave your name and phone number at the Answer Desk during the meeting or mail it to the *Apple Barrel*.

Macintosh

Aztec C	Mike Martin	486-9169
DB Master	Mike Conway	495-2292
Dollars & Sense	Gerald L. Penn	498-0079
	Greg New	568-0290
Filevision	Mike Martin	486-9169
Helix	H. Helm	529-7405
Mac-SIG	Steve Bass	847-4407
MacDraw	Billy Jacobs	480-4410
MacDraft	Billy Jacobs	480-4410
MacPaint	Bill Hailey	682-5147
MacWrite	Bill Hailey	682-5147
Mac 3-D	Larry Stage	358-1105
Megamax C	Chris Flick	363-3153
MGMS CAD	Steve Bass	847-4407
Microsoft BASIC	Bill Hailey	682-5147
Microsoft Chart	Mike Martin	486-9169
Microsoft File	Gerald L. Penn	498-0079
	Greg New	568-0290
	Mike Martin	486-9169
Microsoft Multiplan	Mike Conway	495-2292
	Peter Lemettais	523-1390
Microsoft Word	Virginia Conway	495-2292
OverVue	H. Helm	529-7405
TML Pascal	Grady Beaird	529-8420
Tekalike	Mike Martin	486-9169
XLisp	Mike Martin	486-9169
ZBasic	Tom Dillon	376-6502

Hardware

Abatron Optical		
Character Reader	H. Helm	529-7405
Applied Engineering		
Upgrade Cards	Bruce Sprague	360-1000
Sider Hard Disk Drives	Robin Cox	778-1635
Mac XL	Mike Conway	495-2292
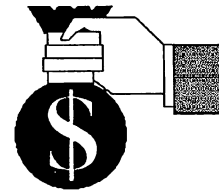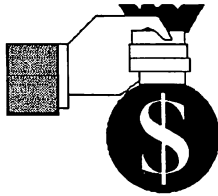
Announcing

FINANCIAL PRO

The newest financial template for use with AppleWorks. Enter the realm of high finance!! By using the **FINANCIAL PRO** spreadsheet templates with AppleWorks you can perform the same calculations that once were only done by bankers, loan officers, and sophisticated investors.

FINANCIAL PRO templates are easy to use. Once the financial data is entered the results are immediately available. By using the powerful AppleWorks spreadsheet, comparisons and "What If" scenarios can be presented.

Here are only some of the 21 templates that are currently included with **FINANCIAL PRO:**

Loan Payment Size & Interest	Depreciation Schedule
Balloon Payment	Depreciation Rate
Loan Payoff	Salvage Value
Mortgage Schedule	Future Value
Initial Investment	Minimum Investment
Annuity (Both ways)	Discounted Cash Flow
Future Value of Regular Deposits	Discount Paper
Maximum Regular Withdrawals	Effective Interest
Minimum Required Deposits	Nominal Interest

Plus the most widely used spreadsheet of all-Checkbook!!

By using the Checkbook template you never have to worry about checks that are out of sequence. Amounts of deposits and withdrawals can be corrected at any time and **FINANCIAL PRO** will automatically recalculate your balance. By using account codes and the sorting feature of the AppleWorks spreadsheet it is easy to categorize your transactions for tax purposes, budgeting, etc.

FINANCIAL PRO will soon be marketed nationally and will cost $37.95.
Now for a limited time HAAUG Members can buy **FINANCIAL PRO** for only $20.00.

Send check or money order to-
        Pro Products
        1179 Bournewood
        Sugar Land, TX 77478
        Phone 713-491-1747 after 6:00 PM

Education SIG

The Education SIG had an exciting June meeting. Fred Fassino presented a review of Apple II music software, Ann Petrillo shared her expertise in software evaluation, and Dick Lee gave a demo of public domain educational programs for the Macintosh.

For those of you that missed it, Ann had some really good information to share on educational software evaluation and she passed out a variety of sample evaluation forms for comparison and discussion. Why tell you about what you missed? Because someday you may be faced with the task of selecting, evaluating, or reviewing educational software for your school or kids and you might remember from this article whom you can turn to for advice.

Okay, I hear our readers asking, "So where is the exciting part already?" How about an offer that can (1) save parents a lot of money on educational software, (2) keep you from taking a loss on educational software the kids outgrow or don't like, and (3) provide you and the kids an opportunity to use more educational software than a family could ever be expected to buy? Thanks to Ann, we found out about a really great program available from the Houston Independent School District (HISD) that makes all this possible. Here is what Ann had to say about HISD's Compubuy Library.

At our last Education SIG meeting, we discussed the sometimes prohibitive cost of educational software. Finding educational software may turn out to be both difficult and expensive. HISD's Compubuy Library offers Apple users an attractive alternative.

The Compubuy Software Library provides you with an opportunity to use numerous educational software packages without buying each one separately. Membership cost is $60 per year. The library includes over 500 software packages and usually two to four copies of each title are available. Two software packages may be checked out for 14 days. A conscious effort is made to carry titles that are not in general use in HISD classrooms.

Library membership is open to the public. One need only stop at the Department of Technology, 5300 San Felipe, between 8:00 A.M. and 5:30 P.M., Monday through Friday. On the first and third Saturdays of each month, the library is open from 9:00 A.M. to 11:30 A.M. For more information, contact the Compubuy Software Library at 960-8888.

While we're on the subject of educational software, Dick put together a collection of programs from the Macintosh public domain that have some degree of educational value. For more information on this disk see the HAAUG Trough article elsewhere in this issue.

Finally, some information on future meetings of the Education SIG. The Education SIG meets at 2:30 and is open to all educators, parents and persons interested in the educational use of computers and the related issues. Y'all come! And now, the coming attractions:

JULY–Computing for Handicapped Students by Bill McClusky.

AUGUST–Software Review & Demo: *Math Blaster*, *Spell It*, and others, reviewed by Glenna Payne.

SEPTEMBER–Software Review & Demo: *Super Pilot*–an authoring program, reviewed by Glenna Payne.

In the Education SIG we place great value in being of service to you and our other members. To meet this objective we need to know what areas of education interest you the most? Come to the next SIG meeting, ask questions, share your ideas, knowledge, and concerns. Help Us Help You!

— *Ann Petrillo*
& Dick Lee

Home Use SIG

As with all evolving things, SIGs (special interest groups) also change. The Home Use SIG has had the guidance of Derrith Wieman since May of last year. Unfortunately, Derrith can no longer fill this position. She, therefore, let the baton fall to me, David Jaschke. I want to thank Derrith for an interesting year of Home Use SIGs. When I asked for a volunteer to help me with the duties and chores of handling a SIG, Mike Stoops agreed. I hope that together Mike and I can present you with an interesting choice of programs for the Home Use SIG.

In the April meeting, I demonstrated the *Micro Cookbook* by Virtual Combinatics. This software allows you to turn your home computer into a cookbook. It provides recipes and lets you enter your own. If your meals are planned ahead of time, the *Micro Cookbook* can be used to print a shopping list for the purchase of the appropriate ingredients. It allows you to modify the ingredients according to how many people are to be served. We did an analysis of how the program would react to a very large serving. When asked to plan a recipe for 160 people, the program calculated that over 40 cups of flour and over 30 teaspoons of another ingredient would be required! This means that to realistically calculate large servings, you would have to recalculate the unit of measurement to a logically larger value.

In the May meeting, Mike Stoops presented the *Newsroom* by Springboard Software, Inc. There was a lot of interest in this product. The program is used to create a mini-newspaper. There is a large selection of graphics supplied; additional graphics disks, called "clip art", can be purchased separately. The *Newsroom* breaks down the process into several areas: Banner, Photo Lab, Copy Desk, Layout, Press, and Wire Service. The banner is that strip that goes across the top of the page. It can contain pictures and text. In the Photo Lab, you take pictures of the clip art graphics called "photos." You can also modify the photo to suit you. At the Copy Desk, stories or articles are written in sections called "panels." You merge your photos with your text here. The newspaper is assembled in the Layout section. The Press is where you print your finished product. Wire Service allows you to transmit any part or all of your newspaper to any other computer that also has the *Newsroom*. A lively discussion ensued as Mike demonstrated how to create a newspaper. Both Mike and I have the *Newsroom* at home. Mike uses it to write letters to friends and in the classroom at school where he teaches. At our home, we use the *Newsroom* to create a family newsletter.

If you are interested in the types of software that can be used on your home computer, drop in and enjoy the Home Use SIG. We hope to see you there for the upcoming meetings when George Pierce will be talking about *Apple-Works*. In later months we will be talking about telecommunications and other subjects of interest to you.

— *DAVID JASCHKE*

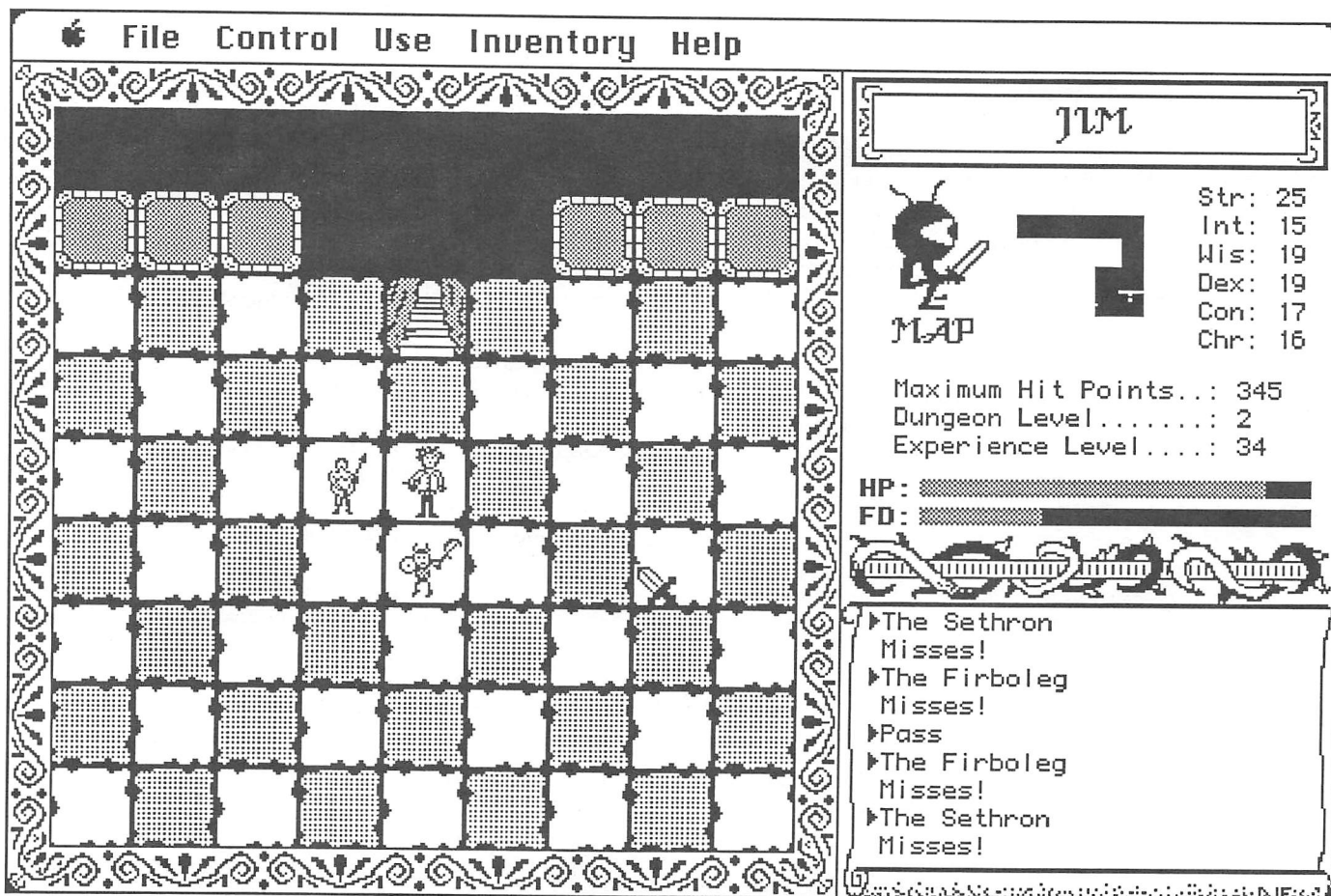# Enter the *Dungeon of Doom*

Computer game addicts might be surprised to find that one of the best games designed for the Macintosh computer is a shareware program written by John Raymonds, *Dungeon of Doom* (version 3.0), which is available in the Mac SIG library. *Dungeon of Doom* is a solitaire fantasy role-playing game much like *Dungeons and Dragons™*. Though Raymonds only requests a $25 shareware contribution, the game is comparable in depth and creativity to commercial games which list for $60 or more.

In Raymonds' game, the player first assumes the role of one of seven different types of characters (see figure 1). Six of the characters have special abilities which correspond to one of seven key objects to be found in the dungeon — armor, weapons, potions, wands, scrolls, rings, and jewels. Each object in the dungeon is assigned a value score. For example, plate armor with a value of +4 protects your character better than chain armor with a +1 score. Knights are able to make educated guesses about which pieces of armor are the best when they encounter them. Similarly, fighters know the value scores of weapons. Sages know scrolls, wizards know wands, alchemists know potions, and jewellers know jewels and rings. A seventh character

type, Jones, has no special knowledge.

Each character also has an set of six personal characteristics — strength, intelligence, wisdom, dexterity, constitution, and charisma. These characteristics, which range from an initial low of 8 to a potential maximum of 25, determine how well the character will perform certain actions. The stronger the character is, the more stuff he'll be able to carry. The smarter he is, the more successful he will be reading scrolls. A very wise character will zap wands better. A very dexterous character will handle weapons better. As his constitution grows more robust, a character's need for food rations

	JIM	
MAP		Str: 25
		Int: 15
		Wis: 19
		Dex: 19
		Con: 17
		Chr: 16

```
Maximum Hit Points..: 345
Dungeon Level.......: 2
Experience Level....: 34

HP:
FD:

▶The Sethron
 Misses!
▶The Firboleg
 Misses!
▶Pass
▶The Firboleg
 Misses!
▶The Sethron
 Misses!
```

File Control Use Inventory Help

Dungeon of Doom Main View

diminishes. Monsters are less likely to attack a character with a high charisma score.

Character types with good initial scores in one area tend to be deficient in others. Thus, while fighters are strong and dexterous, they aren't very smart. Sages are intelligent but not very wise. Jewellers are highly dexterous but also kind of wimpy. Joneses have moderately good initial scores in all six categories, but they have any no special abilities. In choosing a character class, the player has to be prepared to accept trade-offs. I usually picked a Jones because the special abilities of the other types provide few real advantages. Scrolls still fail for sages. Jewellers, alchemists, and wizards must still identify the first of each class of object, even in their area of expertise.

A final characteristic, experience, is at first the same for all character types, zero. A character improves his level of experience in one of two ways, by killing monsters or by successfully reading a gain-level scroll. As your character's experience level increases, he become less easy for attacking monsters to kill. At experience levels lower than, say, 16, characters are pretty vulnerable, so you have to choose fights carefully. After about 20, however, monsters will be easy meat. Getting to experience level 20 is tough; most of the characters you create will be killed very early in their fantasy lives. Fortunately, version 3.0 allows you to "return to life" by restoring a saved game position.

The object of the game is to retrieve the Orb of Canos from the 40th level of the dungeon, and then escape to the surface. To get to the 40th level, you descend through the various levels of the dungeon — exploring the mazes, gathering and using objects, fighting monsters. Each class of objects includes several different types, each with its own uses and characteristics. Figure 2 lists all the objects except weapons that you might encounter in the mazes. Some are more useful than others and I offer some hints about how to use them.

The dungeon creatures are a diverse lot, often with whimsical names, and each can be a formidable adversary depending on the character's current personal characteristics and on weaponry
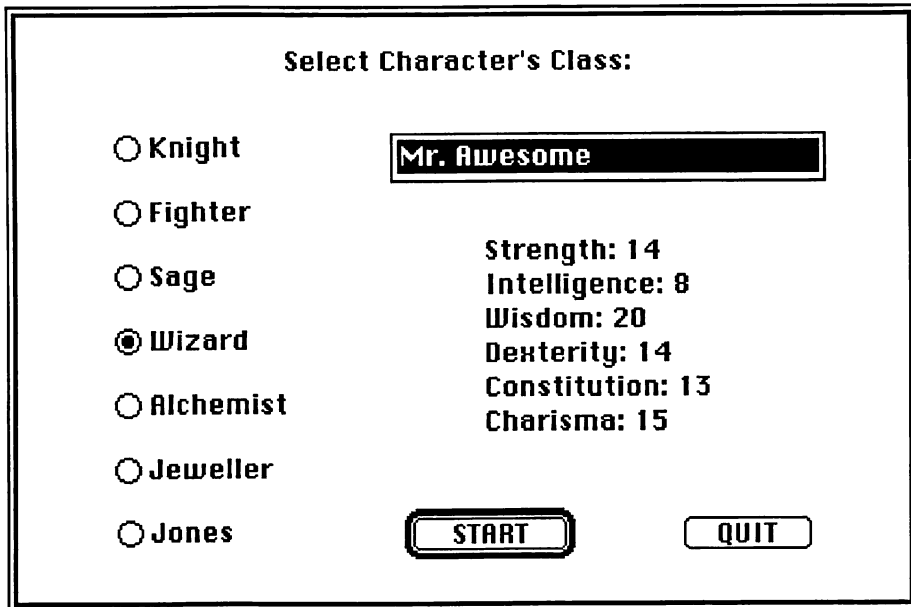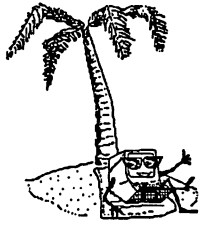

Figure 1. The beginning of a new character

Dungeon of Doom Hints

Here are some hints to make your excursions in the *Dungeon of Doom* less frustrating:

• **Move slowly through the dungeon.** Descending prematurely is a sure way to get killed. Spend some time fattening up on the relatively easy-to-kill Sethrons and Alligogs on dungeon levels 1 and 2.

• **Try to use an identify scroll to identify everything you encounter.** There's a big difference between -1 armor and +4 armor. Unfortunately, identify scrolls are scarce early in the game. At first, you may have to experiment with scrolls, potions, and wands. Use identify scrolls on wands last.

• **Save frequently.** While the game position is saved automatically every time you descend a level, you can save the position as often as you like. Save after every significant event. Save before you read a scroll; if the scroll fails, you can return to the saved position and try again.

• **Fight no more than one monster at a time.** If you get

surrounded, run (or use a teleport scroll); watch your hit points carefully.

• **Use Command-Shift-Caps Lock-4 to print a copy of your Magic Maps.** The black outline maps are not as detailed as the magic maps. Save your Magic Mapping scrolls until you really need them (after dungeon level 20).

• **Use wishing scrolls carefully.** Save before you read a wishing scroll (if it fails, you can return to saved position and try again). Good armor is crucial, so use wishing scrolls to wish for +4 armor if you don't have it already. I wish for chain armor +4 and an elven cloak +4. Also, wish for a +2 regeneration ring. If you get those 3 things, you'll be in good shape. Body armor has a +4 maximum, gloves and helmets have a +2 maximum, and rings have a +2 maximum. You might as well as for the best! Wishing for more wishing scrolls doesn't seem to work.

—CE

Class	Type	Characteristics or use	Length of effect/Notes
Armor	Leather Chain Banded Plate Elven Cloak Helmet Gloves Shield	Better than nothing! Relatively lightweight protection. Heavy Best protection, but heaviest. Lightweight, can be worn on top of other armor. Can be worn with other armor. Can be worn with other armor. Ties up an arm that could otherwise wield weapon.	In wearing armor the trade-off is between weight and effectiveness. Plate armor is the most effective but also the heaviest; leather armor is light but only marginally effective. Watch the value score. Armor with a score of +4 is much better than armor with no score. Elven cloaks can (and should) be worn in combination with other armor.
Scrolls	Gain level Intelligence Wisdom Magic mapping Teleport Enchant weapon Enchant armor Protection Remove curse Identify Joke Words on scroll Amnesia	Increases experience score to next level. Increases intelligence. Increases wisdom. Draws detailed map of current dungeon level. Moves character to another spot in current level. Improves value of one wielded weapon. Improves value of one piece of worn armor. Gives additional protection. Removes curse from armor, rings, and weapons. Identifies unknown item and tells value. Contains joke. Does nothing. Makes you forget all identifications.	Permanent. Permanent. Permanent. Detailed map disappears with next event. Permanent. Permanent. Permanent. Temporary. Permanent. Class permanently identified; value given for specific object only. Ringer scroll. Ringer scroll. Permanent.
Potions	Muscle Strength Dexterity Constitution Poison Invisibility Confusion Levitation Resist fire Resist cold Worthless Speed	Increases strength. Makes you think you're stronger. Increases dexterity. Increases constitution score. Decreases strength. Makes character invisible. Confuses character. Makes character float. Reduces effect of flames. Reduces effect of icy blasts. Does nothing. Makes character move faster.	Permanent. Who knows? Permanent. Permanent. Permanent. Don't drink. Temporary. Temporary. Don't drink. Temporary. Don't drink. Temporary. Effect diminishes as you get zapped. Temporary. Effect diminishes as you get zapped. Temporary.
Wands	Death Lightning Fire Ice Striking Invisibility Sleep Fear Digging Teleport Polymorph	Shoots beam of death. Shoots lightning. Shoots flame. Shoots icy beam. Like hitting monster with weapon. Makes monster invisible. Puts monster to sleep. Makes monster afraid of character. Digs through walls; has no effect on monsters. Moves monster to another part of current maze. Transforms monster into another type.	Most effective but rarest wand. May require more than one zap. May require more than one zap. May require more than one zap. Above wands more effective. Visible monsters are bad enough! Temporary. Temporary. Strength > 20 required to move boulders. Good for temporary relief from attack. Results are too unpredictable. Don't use.
Rings	Regeneration Resist cold Resist fire Slowness Monster X-Ray	Helps character recover faster from injuries. Reduces effect of icy beams. Reduces effect of flames. Makes you slow. Attracts monsters. Lets you see through walls.	Rings only seem to work when they have a value score ≥ 1. In fact, rings with a value < 1 may have an effect opposite from what you expect. For example, a regeneration ring with a score < 1 will retard a character's recovery from injuries.

Figure 2. A complete inventory of items in the dungeon.

and other equipment in his possession. Some monsters can be killed with hand weapons, others are better attacked from a distance with magic wands. A few monsters, like dragons, can zap you from a distance as if they too were carrying a magic wand. These monsters are particularly dangerous because distance strikes cost you extra hit points. Figure 3 lists some of the creatures capable of distance strikes.

The basic strategy of the game seems simple enough — move through the mazes, find and identify objects, kill monsters, accumulate experience and improve characteristics by reading scrolls and drinking potions. Things are not that easy, however. Objects are initially unidentified. You know, for example, that a ring is a ruby ring, but you must use an identify scroll to discover whether a ruby ring is a regeneration ring or a resist fire ring. Further, once you've identified the ruby

Monster	Zaps	Antidote
Icy Whirlwind	Icy Beam	Resist cold potion, ring
Fire Lizard	Flames	Resist fire potion, ring
Dragon	Flames	Resist fire potion, ring
Electric Penguin	Lightning	None
Evil Cleric	Curse	Remove curse scroll

Figure 3. Creatures capable of distance strikes

Rumors:

Death is only a word
The Dark Wizard guards the orb
Potions do not mix well
Money isn't everything
A wizard knows wands
Strength can move mountains
Roni is life
The 40th level is Death
A jeweller knows gems
Monsters don't like stairs
Smart people write scrolls
Jones lives across the street
Taste isn't everything

A bounce can be shocking
Curses can be deadly
A Sethron is 6' 3"
Wisdom guides the wand
A knight knows armor
You can wield a potion
Beauty is only skin deep
Jewellers know rings
Skill wields weapons
Beauty can tame a beast
A fighter knows weapons
Protection can be harmful
A sage knows scrolls

OK

Figure 4. The rumor screen which served as the original instructions.

ring as a regeneration ring, any subsequent ruby rings must be identified with an identify scroll to learn if they are superior to the one currently worn by your character, perhaps a regeneration ring +2. There's danger in using an unidentified object. If you drink green potion, you may discover that it's poison potion. If you zap a bronze wand, you may discover that you've zapped a monster with a polymorph wand, and the monster changes into something nastier.

To make the game even tougher, Raymonds has incorporated random elements. Wands don't always hit, and scrolls often fail. Perversely, wands only seem to miss and scrolls only seem to fail when you're down to your last hit point, and your only hope for escape is a teleport scroll or a fire wand. Other game parameters make careless use of dungeon objects risky. Even if you identify all objects before you use them, you still have to use them carefully. Blasts from several of the wands will bounce off walls. Some of the monsters move very quickly, so a

beam aimed at a monster may hit a reflecting wall instead. Drink too many potions or eat too much food in too short a time, and your stomach may burst. It's not pretty in the dungeon.

I've played both version 1.0 and version 3.0 of *Dungeon of Doom*. I was impressed by the earlier version, but version 3.0 is better in every way. Version 1.0 had a couple of real shortcomings. You were limited to saving only one game position, and, should your character be killed, you were out of luck. You couldn't resume the game at the saved position. Having a character killed after an hour of play is pretty frustrating, particularly when you're experienced enough as a player to be able to build your next character to the same approximate point rather easily. Now, you can return to an earlier saved position even if your character is killed. You can also save several different games-in-progress if you have the disk space (each game file requires 114K).

The other early problem had to do with the fact that were hardly any

instructions for version 1.0. Raymonds initially provided only a set of rumors (figure 4), but version 3.0 features an expanded help menu, which gives much useful information.

Comparing the two versions of the game is interesting. Raymonds has added several new rings, potions, and scrolls. In general, the new objects are ringers which make it tougher to find the really useful objects. The monsters remain more or less unchanged, with a couple of exceptions. One monster, the Fomar, which used to appear somewhere around level three or four, now appears much later, around level 15. Raymonds has also deleted a monster called "the minor devil" and substituted one called "the flying devil." The version one monster had a long pointy tail, horns, and carried a pitchfork; the new monster looks like a large bat. They both behave the same. I wonder if fundamentalist computer owners' complaints prompted this change.

Aside from the improved saved game and help facilities, the most visible change has to do with the way the game board is displayed. In the earlier version, the player was able to see things his character wouldn't be able to see. For example, if your character was "standing" next to a wall on the game board, you could see on the other side of the wall. Now, the only parts of the board you can see are those parts your character would be able to see were he in a real dungeon. Anything out of the character's line of sight or on the other side of a wall is blacked out.

Raymonds has also added a neat automatic mapping facility. As your character passes over a gameboard square, a corresponding pixel in the map area of the status display turns black. It's therefore possible to map a level of the dungeon by carefully maneuvering your character across every square on the gameboard. A white dot marks your character's position. Two other white dots mark the positions of the up and down staircases. You can imagine how useful this automatic map would be in a complicated maze.

There are several other enhancements to the game which don't effect play, but make the game operate smoother. For example, the inventory

windows now scroll, and the lists are now sorted alphabetically. Raymonds has probably spent a lot of time optimizing the code, for everything seems to operate much quicker.

Version 3.0 of *Dungeon of Doom* is a more than worthy successor to the original, an addictive, fascinating pastime, and a fine example of what Macintosh software should be. Diehards may argue that by adding a saved game facility, Raymonds has made the game too easy and unrealistic. This is the old playability/verisimilitude argument with which all game designers must cope.

I have to admit that I've been able to circumvent most of the randomizing elements Raymonds has carefully built into the game's algorithms by carefully saving games and restarting them when things don't work out for the best. However, *Dungeon of Doom* version 1.0 was not always fun to play because it was *so* tough. Of the hundred or so version-one games I began, I only completed two, and only one of those

was really legitimate. In contrast, I've played about five or six version-three games, and two of my characters have escaped with the orb.

As it is, even if you cheat a little, the game requires no small investment of time — say eight hours to get to the 40th level and back. For my money, that makes the game plenty tough enough. Anyway, how realistic is a game with Frobologs, magic wands, and Black Wizards?

— *CLELAND EARLY*

One Minute Over Using the UniDisk 3.5 on the Macintosh

Those of you who know me very well know that I'm rarely afraid to tackle the unknown where computers are concerned. Well, this time I did something pretty scary. I converted my rarely used Apple // Unidisk for use on my Mac(s). To most of you this probably sounds (1) hard to do and (2) something you would never need to do. After all, the Unidisk costs more than the Mac 800K drive so why not just buy the Mac version in the first place?

Many Apple // users also have Macs, and quite a few have also bought the Unidisk for use on their Apple //'s to get away from the age-old problem of too little storage on the //'s 5-1/4" floppies' 143K capacity. From what I've observed, most people with both a Mac and an Apple // eventually ignore the //. So what to do with the Unidisk? Modify it for use with the Mac. It's not all that hard.

Shortly after the Unidisk appeared in the stores, an article showed up in one of the magazines stating that it was possible to modify the drive to work on the Mac. That was the extent of the instructions. Several weeks ago, I took the case off the drive to see what would be required but decided not to do it just yet. I recently got a Mac Plus and found out how nice the 800K drives are. So recently, with Chris Flick and Gary Hamilton watching, I finally did it.

The hardest part was removing the case. There were a lot of screws to take out, but it was not very difficult. Once inside, all that needed to be done was to unplug the interface cable connector plug from the socket on the small circuit board, unplug the short gray ribbon cable from the socket on the main circuit board, and plug the interface cable connector into the socket on the main board. I left the small board with its cable in place, even though it did make it a little difficult to reroute the interface cable. Then I had to find all the screws and get them in

the right holes. Voilà, I was done!

Being prudent once in a while, I decided to test the drive on the Mac 512k, since I'd rather smoke it than the Mac Plus. To my surprise, it worked on the first try, although it would only function as a 400K drive. Whenever previously formatted 800K disks were inserted, the Mac wanted to initialize them. This was because the Mac 512k has not had a ROM upgrade. If it had, it would have used the 800K drive automatically.

I don't recall if it was Chris or my wife who thought of the solution, but someone suggested booting with the startup disk that came with the Apple

HD20. The system on the disk looks for and loads the HD20 driver on bootup and then recognizes the 800K drive and its double sided disks as well as the HD20.

By the way, an unexpected benefit of using the Apple // drive is the long interface cable. It is about twice the length of the cable on my old 400K Mac drive.

— *Mike Kramer*

In the Public Domain:
The HAAUG Trough!

In preparation for the June HAAUG Meeting, I thought it would be interesting to put together a disk of Macintosh Public Domain Software that had an educational theme. First, I wanted to see what was in the 140 disk library I have collected and which reproduces faster than my kid's guppies. Then came the search of local area bulletin boards. Most of all, I had a goal not to create another disk of duplicate programs to clutter up the HAAUG Macintosh collection. Much to my surprise, I was able to come up with 400k of programs. Most are new or at least new versions of programs with use in the field of education. Now some may differ with me on what has educational value so let me state my case.

First came the programs that help teachers, e.g., gradebooks and statistics programs. Then it was programs that help students remember, e.g., Flashcards complete with several subject files. Next were demo programs that demonstrate some of the potential the Mac has for helping students learn, e.g., Anatomy, molecular orbitals and earth plots. It's these programs that I have selected to review as pick of the litter.

Anat.Bin
    Version 0.91, 18k
    Application — A demo of a mid-leg skeletal anatomy lesson and test, it has exciting educational possibilities for many other subjects. Additional information about Anatomiser is available from the author, Stephen Durbin V.M.D., Ph.D., BioMedical Engineering & Science Institute, Drexel University, Philadelphia, PA 19104 or through Compuserve # 76074,55.

µEarthPlot
    Version 2.0, 53k
    Application — Construct interesting projections of earth by changing longitude, latitude and altitude. *µEarthPlot* is by Michael Peirce with version 2.0 improvements by Marsh Gosnell.

Flashcard
    Version 1.1, 22k
    Application — A repetitive-enforcement learning aid, designed to be similar to flashcard decks commonly used to teach mathematics and other skills. A really neat program that allows you to build your own flashcard decks for any subject. This program comes with example Flashcard Decks for the elements, gland/hormone function, Morse code, and several others. Program and documentation written by Bob Ellison of the Rice Macintosh Development Team.

Grades
    Version 0.96, 32k
    Application — A gradebook; handles grades for up to 24 items (test, homework, etc.) and allows weighed items. Link-grade, Grader & Grade ".text files" required to run. Created by Jerry Keough, Department of Mathematics, Boston College.

Orbital Mixing
    12k
    Application — Depicts Molecular orbitals derived from two 2p orbitals overlapping in PI manner. For organic chemistry students with a modern text and more theoretical interest than those of us with a need to get the chemicals out of the reactor and into the pipeline. Written by John R. Meier. Another good example of the Macintosh's use in education, this time in visualizing a complex concept. Would be better if there were any documentation.

Regres
    33k
    Application — (Multiple Linear Regression) Regression analysis, if you understand it you shouldn't have much trouble with this application...if you don't, forget it! *Longly* is a support file the program creates as an example. More information can be obtained by opening *Longly* as a *MacWrite* file. Program by Steve Brecher, Software Supply.

Parametric Statistics
    Version 2.0, 40k
    Application(SHAREWARE)
    Another math/statistics application that requires understanding the subject to figure out. Interesting to statistics buffs. A t-Test, One Way ANOVA & Linear Regression Program by Cary N. Mariash.

— *DICK LEE*

Coming to the October Macintosh Main Meeting:

Andy Utter of Kent-Marsh Limited demonstrates the *MacSafe* file security program

Alsoft Software with *Disk Express*. This product speeds up disk access by placing files on contiguous blocks

Macintosh Programming: A Step-By-Step Approach

Step II—The Main Event

In this installment of programming the Macintosh we will begin to set up a program skeleton that we can then fill in with the specific application. Again, we will be talking about using Pascal as the programming language, but the concepts will be similar regardless of the language used.

To program the Macintosh computer, one must memorize the following statement: **The Mac is an EVENT driven computer and I must treat it like one.** In essence, programming the Mac is very simple; all the program does is to interpret and act on a stream of events in a particular way. A generic program consists of an initialization section, a section that gets the event, and a section that processes the event and does something.

So now that I've talked about events for a paragraph, it's probably worthwhile to define what an event is. An event is any user generated activity; it can be the press of a key, the movement or click of the mouse, an application generated event, or the insertion of a diskette into the computer. There are two other events that are indirectly generated by the user; the window events (activation, deactivation, and update events) and the null event (e.g. no event).

Events are accumulated in an **Event Queue.** Events are handled in a FIFO mode; First In, First Out. Each event is not a data point but in a collection of data called a record. An **Event Record** contains quite a bit of information. The event record contains information about the type of event, where the mouse was when the event occurred, when it occurred, the event message (what key was pressed, which disk drive the disk was inserted into, etc), and modifier information (option, command, mouse, shift or caps lock keys up or down). Your program can use any of the information in the event record.

To get the event from the event queue, one can use the toolbox call **GetNextEvent** to get the next event. The form of the command is:

Function GetNextEvent (eventMask: INTEGER; VAR theEvent: Event-Record): boolean;

The eventmask is a integer value that tells the computer what event you are looking for. This allows you to look for a particular event before any other. If an event is chosen, the function returns a TRUE value and the

Figure 1. Program Design Flowchart

variable theEvent contains the event record. If no event is chosen, a FALSE value is returned and theEvent is a null record. If a true value is obtained, that event is removed from the event queue and the next event is brought to the front of the queue.

Once you have the event, you can use the information directly or modify it. An example of modifying an EventRecord would be to take the event corresponding to pressing a return key and change it to a mouse click in an OK box of a dialog box. Then the modified event would be handled by the application as if it was a mouse click.

Our program design is flowcharted in Figure 1.

Next Time: All the work in the program deals with the section "Do Something with the Event." Our next article will emphasize the menu bar; we'll create an event and use our eventhandler to look for menu bar events and do something with them.

Assignment: Read the Toolbox Event section in *Inside Macintosh* and the "Keeping Up with Events" chapter in the *Macintosh Revealed Volume II*. After you have completed this task, examine all the possible types of events that your program will need to use. Based on your storyboard that you put together from the previous assignment, figure out how each each event will impact on your program and how you will use those events to move to the various parts within the program. You should then put together a flowchart to expand the "Do Something with the Event" section of the flowchart. Use as much detail as you can. At this point you are ready to start generating actual program code. We'll begin doing this next time.
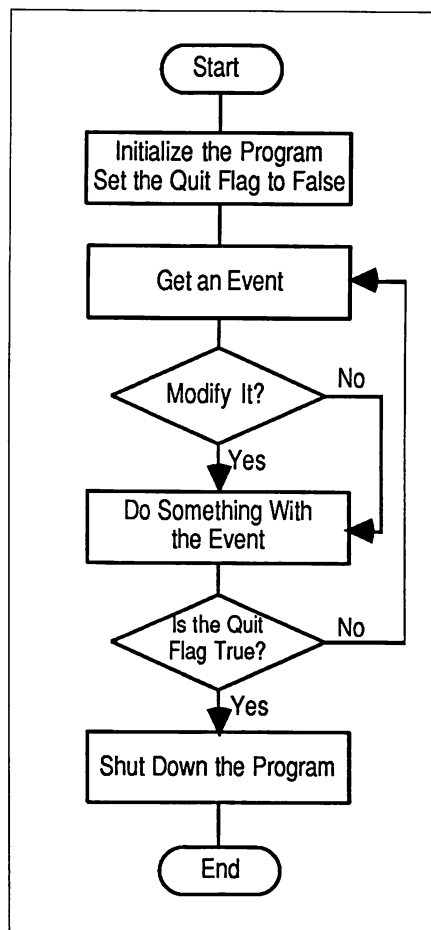
— *MIKE CONWAY*

```
Mem # 523              Expires: 10/86
Robin A. Cox
5401 Rampart #662
Houston, TX  77081-1329
```

Meeting Dates
1986-87

—————— • ——————

September 20, 1986
October 18, 1986
November 15, 1986
December 20, 1986
January 17, 1987
February 21, 1987
March 21, 1987
April 18, 1987