Aurora Area Apple Core Newsletter

Published Monthly by the Aurora Area Apple Core, a Club for users of Apple Computers

Volume VII Number 2 February 1991 Issue

SPRING IN FEBRUARY?

It sure feels like spring around here as I'm typing this edition of the AAAC Newsletter. The evening temperature is 40 degrees and tomorrow is to be in the upper fifties. WOW! wait till next week though, probably back to winter. I hope you are enjoying this lovely weather for as long as it lasts. At least the heating bill won't be as high during this interval.

ANOTHER ZIP CHIP BUST

Well, seems that another Zip Chip bit the dust. Gary Senesac lost his a little while back and at this time he hasn't received his replacement. It seems that the three of us who purchased the Zip Chip when they first came out have had at least two replacements so far. I'm on my third chip and with my fingers crossed it seems to be working just fine. Is there any other members who have Zip Chips? Let me know if you have one and how they are doing. I'm curious as to the what the 8 mHz chip is like, mine is the 4 mHz version.

HARDWARE BREAKDOWN

At the meeting in January we had a problem with the hardware and our great Secretary was unable to give his presentation. I've checked and to date the equipment is in working order and George Murphy will present the program OUTLINER from the TimeOut series. Yea! George. Also George is a Beagle Buddy and there is an item of his later in this issue of the Newsletter.

NEXT MEETING

Mark your calender for Thursday, February 14, 1991, at 7:00 PM in the basement of the former Suburban Bank Building, in the NorthGate Shopping Center, on Route 31 in Aurora. Come to the meeting and bring a friend and see George Murphy give his presentation. Also don't forget to visit Lois and Jack Hammond and purchase a couple of raffle tickets. Also there will be friendly, smiling Jon from H & H Enterprises with goodies for your computer.

A BIG THANK YOU

To George Dybowski for the fine series of articles that he has made available for our issues of the AAAC Newsletter. If any of the members have any articles that they have available, it would be appreciated and also rewarded with free disks from the AAAC Library.

INTERESTING?

While glancing through the Sunday edition of the Chicago Tribune, the Business Section, I came across my favorite section, the Computer News and Ads.

One article mentions that Apple is asking the FCC for permission to use the space between 1850 and 1990 MHz for all computer companies to use for networking systems. That portion of the band width is already used by several industries, including railroads and utilities, for their in-house communications.

If the FCC approves Apple's petition--which may take years--then the cables and connectors that are the backbone of a typical office computer network could be eliminated. In their place, each computer would contain a transceiver that would transmit and receive radio signals on the assigned frequency over an effective range of about 150 feet.

Bill Stevens, manager of wireless communitions research at Apple, said he did not expect such radio-based systems to completely replace those using cable, since cable systems will probably always be cheaper and better-performing.

He said, though, that radio networks could be useful in situations, such as schools, factories and many offices where flexibility and mobility are prized.

MACINTOSH INFO

Ye Olde Editor has had his eyes on the new Macintosh line of computers from Apple but there are several questions that need be answered. Would an upgrade from the faithful old IIe to a Mac be better than an upgrade to a IIgs? From experience, the Apple line seems more appropriate than shifting to the IBM clone, as a few of our members have done. Their reasoning is that they are using that type of computer at work and another reason is that the programs are not available for the Apple that are on the market for the IBM series. I like the Apple programs much better than the others because they much more user friendly. I use both systems, Apple and IBM so I can compare systems. The IBM is at the store where I work and naturally the Apple is at home where I am doing this Newsletter. Anyhow....... Here are a few specs from the new Mac line. It may be of interest to you.

Macintosh Classic

An updated version of the most popular Macintosh compact design, the Macintosh Classic personal computer runs applications up to 25 percent faster than the Macintosh Plus and can easily handle word processing, spreadsheet analysis, and a wide variety of other business, home, and education applications. The Macintosh Classic includes:

8-megahertz 68000 microprocessor, 1 or 2 megabytes of RAM (expandable to 4 Megabytes), Apple Super-Drive floppy disk drive (for reading from and writing to Macintosh, MS-DOS, OS/2, and ProDOS floppy disks), Optional internal 40-megabyte hard disk, Keyboard, Integrated 9" monochrome monitor, Six built in ports, and all-in-one, integrated desigh that's easy to set up and easy to move.

Macintosh LC

The Macintosh LC personal computer offers a solution for people who want the advantages of Macintosh with the benefits of color at the most affordable price. The Macintosh LC is an affordable way to take advantage of color, graphics, and expandablility for managing small businesses, doing work at home, or running educational applications. The Macintosh LC includes:

16-megaherty 68020 microprocessor that enables the Macintosh LC to run applications up to three faster than the Macintosh SE computer, 2 megabytes or RAM (expandable to 4 or 10 megabytes), Apple SuperDrive floppy disk drive, internal 40-megabyte hard drive, support for three Apple monitors, offering customers a choice of color or monochrome displays without adding a video card, sound input capability built in, seven built-in ports, and with the addition of an Apple IIe Card, the Macintosh LC allows you to run Apple IIe programs.

Macintosh IIsi

The Macintosh IIsi personal computer provides the high performance, flexibility, and expandability of a Macintosh II class of systems at a very affordable price. The Macintosh IIsi computer includes:

20-megaherty 68030 microprocessor, 1 megabyte of RAM (expandable to 17 megabytes, Apple SuperDrive floppy disk drive, on board video support for four monitors (including the portrait display), allowing you to select from a variety of color and monochrome monitors from Apple Computer and third-party companies, sound in capbility built in, eight built-in ports, and expansion slot for NuBus or 030 Direct.

FOR SALE

Apple IIe enhanced, 1-Meg Ram Works 3, DuoDisk 5 1/4", Unidisk 3 1/2" disk drives, Mouse and Mouse Paint, Green Monitor, 2400 Baud Modem, lots of software. $1,000.00 or best offer. Call Jack Hammond at 708-892-9621.

INFO AND/OR PARTS NEEDED

Glen Schultz at 708-554-1118 is looking for a DuoDisk drive, ThunderClock *Software and an* external cooling fan for his Apple.

APPLEWORKS SPREADSHEET

THAT'S A MIGHTY BIG "IF"

The ability to reason is generally considered to be the edge we humans have over animals. The relationship between calculators and computers is very similar. Calculators can only act on the numbers entered in certain, predefined ways. They cannot make decisions based on the input. Computers however, have the ability to take information given, test it, and perform different operations on it depending on the results of the test. The most common way a computer does this is with an "IF" command. In FORTRAN it's called an IF-THEN statement. In spreadsheets, the form is usually @IF(test, result if true, result if false). Many other spreadsheet commands are variations of the IF command: CHOOSE, LOOKUP, MAX and MIN. These commands perform tests as opposed to performing a math function of some kind like @SUM, @AVERAGE, and @COUNT.

Let's look at the components of the IF statement.

```
@IF(Test, result if true, result if false)
```

"Test" is the part of the statement where the question is asked, and depending on your programming environment, you can ask darn near anything. In most cases, a mathematical comparison is used: A>B, A+B, A=0, A*3<B*5, etc. "A" and "B" can be values, equations, and in spreadsheets, are most likely to be cell references. When calculated, the IF statement will come to one of two results: "True" or "False", that is, the statement or equation is either logically true or false. If the result is "true", the first statement after "test" will be performed, and the second statement will be ignored. If the result is "false", the first statement is ignored and the second is performed.

An example: Compare cell A1 to cell B1. If A1 is larger than B1, multiply A1 times B1. If A1 is not larger than B1, print zero. The IF statement would be written as follows:

```
@IF(A1>B1,A1*B1,0)
```

A variation: If A1 is larger than B1, print A1. If B1 is larger than or equal to A1, print B1.

```
@IF(A1>B1,A1,B1)
```

A very similar test could also be written using a variation of the IF statement:

```
@MAX(A1,B1)
```

The MAX command will automatically choose which is larger and display it. This is especially useful when comparing more than two values. More on MAX later.

AppleWorks 3.0 can use text in equations. The above IF statement might be written as:

```
@IF(A1>B1,"A1 is larger than B1","B1 is larger than A1")
```

But what if A1 equals B1? The IF statement would determine this as "false" because A1 would not be more than B1. The situation may arise that if A1 equals B1, a different result needs to be identified, for example, zero. The IF command only allows two choices, "true" and "false", BUT, instead of displaying a result, another test can be made. That is, additional IF statements can be built in. This is called "nesting", meaning one IF statement is "nested" inside another. For example:

```
@IF(A1>B1,A1,@IF(A1=B1,0,B1))
```

If A1 is greater than B1, A1 is printed. If A1 is not greater than B1, the "false" result will perform a second test: is A1 equal to B1. If "true", zero is printed, otherwise B1 is printed because both tests come up "false". The above equation can be written several ways and still return the same results:

```
@IF(A1>B1,A1,@IF(B1>A1,B1,0))
@IF(A1>B1,A1,@IF(A1<>B1,B1,0))
@IF(A1=B1,0,@IF(A1>B1,A1,B1))
```

Same question, different structure and syntax, same answer.

This type of nesting can be repeated within an equation until you run out of room in the cell. In AppleWorks, that's 80 characters. This would seem to limit the complexity of an IF statement, but there are ways to get past this for virtually unrestricted IFing (sometimes without even using an IF statement!).

The most common need for this type of statement is getting a result that varied over a range of conditions. The best example of this is a commission structure. For example, there are six commission levels:

Sales Range	Commission %
0 - 999	10%
1000 - 1999	11%
2000 - 2499	12%
2500 - 3999	13%
4000 - 4999	14%
5000 or more	15%

The amount of sales is in cell A1. A nested IF statement would look like this:

```
@IF(A1<1000,.10,@IF(A1<2000,.11,@IF(A1<2500,.12,@IF(A1<4000,.13,@IF(A1<5000,.14,.15)))))
```

This equation won't fit in the cell because it is longer than 80 characters! It is also a very tedious equation to develop, type in and edit. What will we do? IF's specialized cousin LOOKUP will help us out.

LOOKUP will take a value and look it up in a table, select the appropriate result, and bring it back to the equation. To use the problem above, a table is created in cells B1 through C6:

	B	C
1	0	.10
2	1000	.11
3	2000	.12
4	2500	.13
5	4000	.14
6	5000	.15

The LOOKUP equation to find the correct commission percentage would be written as follows:

```
@LOOKUP(A1,B1.B6)
```

The value shown in cell A1 will be looked up in cells B1 through B6. This is the equivalent action of all those nested IFs shown above. When the proper value is found, the corresponding value in column C will be returned. If A1 was 2833, the value .13 would be returned. There are other advantages to this type of equation besides getting around the 80 character limitation. They are: much easier updating if the commission values change, easier viewing of the actual commission structure, less cluttered equations and the table can be accessed from elsewhere in the spreadsheet.

The equation can be modified to calculate the actual commission earned:

```
@LOOKUP(A1,B1.B6)*A1
```

CHOOSING RATHER THAN LOOKING UP

Another specialized cousin of the IF statement, is CHOOSE. It shares characteristics of both IF and LOOKUP and allows you to do a lot of IFing with a small number of keystrokes. The format of CHOOSE is:

```
@CHOOSE(value to lookup,result if 1,result if 2,result if 3,...,result if n)
```

What CHOOSE does is lookup the value (or value of the reference) in the rest of the CHOOSE statement, unlike LOOKUP which looks up the result in an external table. The first result in the CHOOSE statement will be chosen if the lookup value is 1. Result 2 is chosen if the lookup value is 2, and so on). (In BASIC and other languages, the ON x GOTO and ON x GOSUB statements work on the same principle.) The best example of this would be one that we all have do every year, choose the correct standard deduction based on our filing status for our federal income taxes:

Status	Status	Standard Deduction
Single	1	3,250
Married - joint	2	5,450
Married - separate	3	2,725
Head of Household	4	4,750
Widow(er)	5	5,450

Cell A1 will contain the status #, and equation will look like this:

```
@CHOOSE(A1,3250,5450,2725,4750,5450)
```

Note: the above table is what the lookup table for the LOOKUP command would look like. In this case, either the CHOOSE or the LOOKUP command could be used here.

If the value of A1 was not between 1 and 5, an #NA would be the result. When using the CHOOSE command, it is wise to put it in an IF statement that will pre-check the choosing value:

```
@IF(A1>5,@IF(A1<1,@ERROR,@CHOOSE(A1,3100,5200,2600,4550,5200)))
```

This forces the equation to show that there is an error in the choosing value.

Here's a trick to shorten some CHOOSE statements (if applicable):

```
@CHOOSE(A1,1000,3000,8000,4000,9000,15000,35000,2000,19500)
```

can be shortened to:

```
@CHOOSE(A1,1,3,8,4,9,15,35,2,19.5)*1000
```

saving 26 characters in this statement. Something to remember if you want to save a few bytes in the spreadsheet or cram more equation into a cell.

Warning about CHOOSE and LOOKUP: You can only use one @CHOOSE or @LOOKUP statement per cell (equation). I don't know why, but AppleWorks can't deal with more than one at a time. In other words:

```
@CHOOSE(A1,2,5,7,4,3,8)+@CHOOSE(B1,66,45,3,78,999,123)
```

will give you an error beep even though there is nothing illegal in the equation.

MAX and MIN

So all they can do is pick the biggest and smallest numbers, big deal. These two commands are deceptively powerful. They act in the same manor as LOOKUP and CHOOSE by automatically combining many IF statements into one small command.

The most common uses for MAX and MIN are to find the biggest and smallest values in a range:

```
@MAX(C4.C25)
```

or to compare values:

```
@MIN(A3,A4)
```

More than two cells or values can be compared:

```
@MAX(B2,C5,A4*6,D1^2,3.14.1596,E1.E9,A8/G4)
```

The most useful technique I use with MAX and MIN is to take a very cluttered IF statement and simplify it. Example: Look at the last few lines on a 1040 tax form. The math they want you to perform is: On line 64 - if line 63 is greater than line 55, subtract line 55 from line 63, otherwise enter zero. This is followed by a similar calculation on line BB - if line 55 is greater than line 63, subtract line 63 from line 55, otherwise enter zero. If the equations were composed as stated, they would look something like this:

```
Line 64:  @IF(L63>L55,L63-L55,0)
```

We can do better than that:

```
@MAX(L63-L55,0)
```

The logic of the MAX statements perform the subtraction and then compares the result to zero. If L63 is not greater than L55, then L63-L55 will be a negative number, and zero will be the result. Much neater and cleaner.

Of course you can also nest these statements with IFs too. Example: multiply cell A1 by cell B1. The result is not to exceed 5000 unless cell C1 is more than 30:

```
@IF(C1>30,A1*B1,@MIN(A1*B1,5000))
```

And you can get a whole lot more complex than that!

SUMMARY

As you can see, IF, LOOKUP, CHOOSE, MAX and MIN can allow your spreadsheet to do some dazzling tricks. Just remember, the most difficult part of using these commands is getting them to give you the correct answer in all possible combinations of true and false results. Debugging these statements can be tricky, but when you get them working right, the result can be amazing.

A final word of caution. These words are from the Lotus 123 manual:

"Allowing a computer to make decision means there is the possibility of an unanticipated decision. Programmers call these unanticipated decisions 'bugs' ."

George Dybowski

A MESSAGE FROM OUR PRESIDENT

"HAPPY VALENTINE DAY"

A MESSAGE FROM OUR TREASURER

"Are your dues due? See me at the meeting or mail your dues to me via the Post Office Box listed on the address side of this Newsletter"

BEAGLE BUDDY NOTES:

(1) I HAVE RECEIVED 4 NEW UPDATES FROM BEAGLE BROS. THEY ARE: TEXT TOOLS V 1.2, REPORT WRITER V 2.5, SUPERFORMS AND TELECOM V 1.2. THEY ARE AVAILABLE IF YOU NEED THEM UP GRADED.

(2) A NEW RELEASE BY BEAGLE BROS. IS PICK FONTS WHICH MAKES SUPERFONTS EASIER TO HANDLE. BEFORE PICKFONTS, USERS HAD TO TYPE EACH FONT NAME AT THE TOP OF A WORD PROCESSOR FILE. FONT NAMES CAN BE HARD TO REMEMBER AND DIFFICULT TO SPELL. PICKFONTS ELIMINATES THAT PROBLEM BY PRESENTING THE USER WITH A LIST OF FONTS TO CHOOSE FROM. ONE OR MORE FONTS CAN BE CHOSEN FROM A LIST, AND THE FONT NAMES AND NUMBERS ARE AUTOMATICALLY ENTERED AT THE TOP OF THE WORK PROCESSOR FILE FOR EASY SELECTION. PICKFONTS IS NOW AVAILABLE ON THE SUPERFONTS DISK, VERSION 3.1 AND I AM AUTHORIZED TO UPDATE YOUR SUPERFONTS DISKS AS IF IT WAS A REGULAR UPDATE. THIS SYSTEM WAS DEMONSTRATED AT OUR DECEMBER MEETING.

(3) ALSO DEMONSTRATED AT THE DECEMBER MEETING WAS THE "LIBRARIAN" FUNCTION OF THE TIMEOUT POWERPACK PROGRAM. THIS WAS IN RESPONSE TO A QUESTION RAISED IN A PREVIOUS MEETING. THE DEMO SHOWED HOW QUICK AND EASY IT IS TO READ FILES FROM SEVERAL DISKS AND THEN PRINT OUT THE LIBRARY LIST OF THE FILES ON EACH DISK IN FILE ORDER.

(4) THE WINNER OF THE AW 3.0 COMPANION DISK THAT WAS SENT TO US WAS MARIANNE LIEBEZEIT.

(5) BEAGLE BROS SENT ME A COPY OF A DATA BASE LISTING ALL OF THE LATEST REVISIONS OF ALL OF THE TIMEOUT SERIES. I PRINTED IT OUT AND SINCE IT TAKES 6 PAGES I WILL NOT INCLUDE IT IN THE NEWS LETTER. IF YOU WANT TO KNOW IF YOU HAVE THE LATEST VERSION SEE ME AT THE CLUB MEETING.

THAT'S IT FOLKS

GEORGE MURPHY

Aurora Area Apple Core

President	Vice President	Secretary	Treasurer
Charles Davis	Glenn Shields	George Murphy	George Dybowski
879-2321	552-8565	357-0759	879-9253

Newsletter Editor	Software Library
Bill Swiss 554-1225	Art Shaff 898-9528

Directors: Gary Senesac 879-7724 Jan Sprague-Williams 892-8746
Youth Department,

The Aurora Area Apple Core meets on the **Second Thursday of each month** at 7:00 P.M. in the basement of Suburban Bank, in NorthGate Shopping Plaza, on Route 31, Aurora. Membership is open to all persons interested in the Apple Computers and in developing their use through the exchange of ideas and information. The public is cordially invited and bring a friend. If you have any questions about the club or programs please phone one of officers listed above at a reasonable time.

This Newsletter is copyrighted and published monthly by the Aurora Area Apple Core. Reproduction in whole or part is prohibited unless proper credit is given to the author and this publication. Opinions expressed herein are not necessarily those of the Aurora Area Apple Core, which assumes no liability for articles herein.

Aurora Area Apple Core Newsletter is an independent publication not affiliated or otherwise associated with or sponsored by Apple Computer, Inc. The opinions, statements, positions and views stated herein are those of the author(s) or publisher and are not intended to be the opinions, statements, or positions or views of Apple Computer, Inc.

Aurora Area Apple Core "Newsletter"

Aurora Area Apple Core
Post Office Box 2901
Aurora, Illinois 60507

Aurora Area Apple Core "Newsletter"