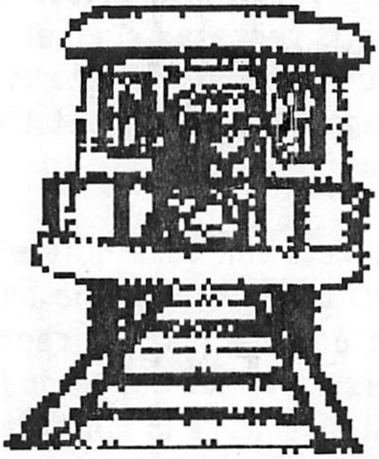
As you will see, each of the three tests result in vastly different times to sort. Re-running the identical tests, but with a different randomized array would likely result in slightly different times. Also shortening or lenghtening the list will make a difference in the relative times. Some sorts work very well on long lists but fall on their faces on short lists. Others like the most used sort of all, the Bubble (or Exchange) sort, work well on short lists but show their weaknesses on long lists. We'll start our investigations with the bubble sort.

The bubble sort is so called because if one watches the the sort in action on some type of graphics, the high value items seem to bubble up to the top as each pass is made. This sort is most likely as popular as it is because it is simple and short. The idea in a bubble sort is that the first item is used to compare with the second. If the first is larger, the two of them are exchanged (hence the other name for this type of sort). If the second is smaller than the first, the first is compared with the third and so on until a larger item or the end of the file is found. If the end of file is found, the item is inserted there.

If a larger item is found and exchanged, it becomes the item for comparison until, as before, a larger item or the end of the file is found. Each time an exchange is made, a flag is set indicating that an exchange has taken place. The routine is repeated over and over again until a pass results in no exchanges. At that point the list is sorted.

This process can take quit a while to execute. A run of the bubble sort on an array of 1024 random integers just took 980 seconds to complete. That is 16 minutes 20 seconds. On an inverted sort, a bubble sort took 1462 seconds (24 min. 22 sec.) On a partially sorted list, this sort took 483 seconds (8 min. 3 sec.). I don't know about you but somehow I expect better of a computer. It still may have been faster than I would have been manually, but it verges on making me think I may have been able to better it. Let's look at some other sorts.

```
PROCEDURE bubble
PARAM low,high,ndata(1024):INTEGER
DIM count1,count2,temp:INTEGER
DIM flag:BOOLEAN
FOR count1=low TO high
flag=FALSE
FOR count2=1 TO high-count1
IF ndata(count2)>ndata(count2+1)
THEN
temp=ndata(count2)
ndata(count2)=ndata(count2+1)
ndata(count2+1)=temp
flag=TRUE
ENDIF
NEXT count2
EXITIF flag=FALSE THEN
ENDEXIT
```



THE LIBRARY CAR

by Al Wagner

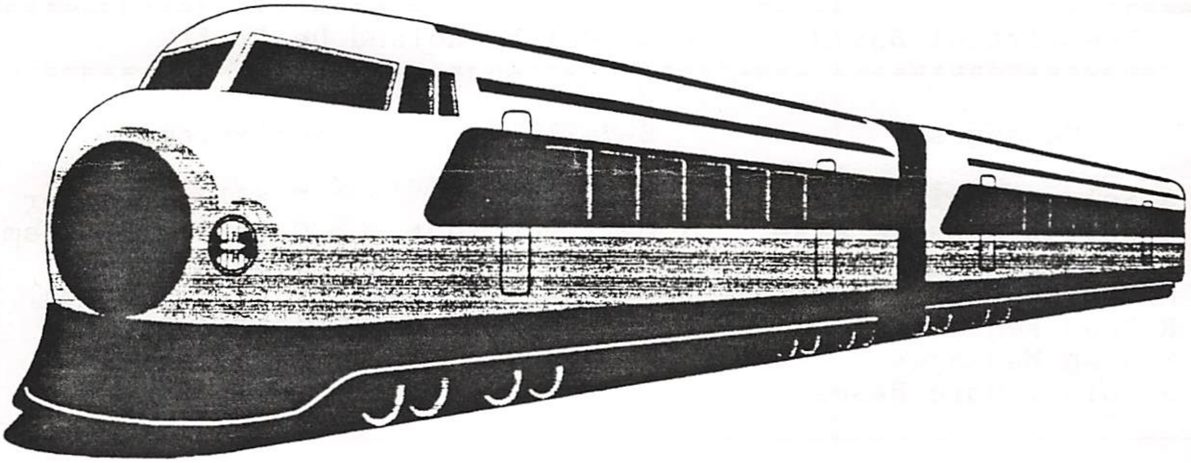
Welcome to the January 1993 edition of the Library Car. I've got the pot belly stove fired up over here so pull up a box, crate, or whatever you can find that will hold you off the floor and we'll talk about sorts this month. I know, some of you are groaning already, "Not another round with sorts!" Well, I for one have never looked into a variety of sorts and never really looked at even one but in passing, so bear with me on this. I am not going to give a line by line explanation of each sort. Try to mentally run the programs. As you'll see in one of the explanations below I had to empty the change from my pockets and sort it using the routine in the program to make sense of how it worked. Try it! Its not as hard as you think and once you analize a program this way, you'll really understand how it works. Keep in mind they all DO work. If any of you don't understand how a particular sort works and are interested, I am willing to re-cover any or all of them as needed. For now though, a brief explanation and the listing should suffice. The listings this month do not have offset numbers as I did not have to run them through the Basic09 compiler. Near the end of the article the reason for this is explained and credits given.

A couple of notes on the methods of testing are in order here. In each case an array of 1024 randomized integers was used to test the sorts. The same array was used for each sort tested. Three tests were performed on each sort. The first was to sort the randomized array from smallest to largest. This test works a more or less "real world" condition of a list that just kinda happened. The second is worst case and is unlikely to just happen in reality. The sorted array was inverted (made to read largest to smalest) and then re-sorted to read smallest to largest. The third test was on a partially sorted list. The sorted array was left intact, but the number 1 was added above the largest number.

This simulates another "real world" problem where a sorted list exists but more data was added to the end that needs to be sorted to somewhere further up in the list.

fact in mind! A used XT is by no means an upgrade.

Rick Hengeveld



Get on board for the March 25th meeting! 7:30 PM!

FIND YOUR WAY AROUND THE MAVERICK!

Sysop's guide to VBBS software

Several weeks ago I attempted to demonstrate our latest BBS software at our regular PJCCC meeting. Since that time I've noticed that a good number of user's are still having problems understanding the system. This is logical since the present system offers many more user options than the former software. This is my effort to clarify the systems menus and features, it would be a good idea to keep this guide handy until you are totally comfortable with the BBS and its functions.

After your connection to the Maverick you will be presented with a systems statistics screen. This simply informs a user as to how busy the BBS has been, total calls, uploads etc. After this the system automatically checks to see if you have any unread E-Mail. You may at this time read your personal mail or skip over it and read the mail at a later time. After you read your mail please delete it by pressing "D" at the prompt. This is important as it preserves drive space that would otherwise be taken up by outdated messages. Once these hurdles are cleared you will be taken to the main menu. Here is a sample of the main menu and a short explanation of its features.

```
!==============================================================!
!   The Virtual BBS/NET 6.00 Copyright Roland De Graff         !
!==============================================================!
!                                                         !
!      Messages              E-Mail          SubSystems        !
!                                                              !
! $ Select Topic Area     M Mailbox Scan       T File Transfer !
! C Select Message Base   Y Mail you sent      O Online program!
! P Post a Message        E Send Email         D Defaults      !
! N New Messages          F Feedback to Sysop  V Voting Booth  !
! R Read sequential       Q MultiMail          A Autoposts     !
! S Scan Messages                              B Bulletins     !
! J Join/Ignore Bases                          ! QWK Functions !
!______________________________________________________________!

L Page Sysop      U User Listing   W Who's online    X Net Info
I System Info     K Today's Calls  Z Teleconf.       G Logoff
================================================================
```

As you can see all the message functions are located on the left column of the menu, we'll deal with these first.

$ Select Topic Area: This BBS currently deals with and carries messages about, Computers (How novel!) We could though, create other topic areas pertaining to any other subject. At this time we carry only 1 topic, listed under the heading "Main". Since there is only one Topic you should never have to use this option.

C Select Message base: Under the main topic you can find up to 12 individual message bases. Selecting "C" from the menu will list these bases by name and number. After these base listings appear you may jump directly into any one of them simply by pressing its number at the prompt.

```
NEXT count1
END
```

The next sort is called an Insert sort. To follow this one, I took out some change, placed it on the table in a random order and followed the program, sorting the penny, nickel, dime, quarter and half dollar. Its not too easy to follow but, keep in mind that count1 is one less than count2 until the bottom of the REPEAT/UNTIL loop is reached and then it is 2 less than count2. Count2 is used to pickup the temp item before the R/U loop. Come on now, get out the change and sort it.

This DOES make sense! Basically the program keeps moving larger items further along the list until it meets with an item that should be moved back in the list. If so, it deposits the item it has been moving and then moves the smaller item back up the list. This sort took 488 seconds (8 min. 8 sec.) on a randomized array (a pretty good improvement), 956 seconds (15 min. 56 sec.) on the inverted list (not too shabby), but hold on to your hats, it took only 4 seconds to work a partially sorted list! Of all the sorts tested, this one worked the best on a partially sorted list. That makes this sort a good candidate for service where the primary job is to insert new data into an already sorted list.

```
PROCEDURE insort
PARAM low,high,ndata(1024):INTEGER
DIM count1,count2,temp:INTEGER
FOR count2=low+1 TO high
count1=count2-1
temp=ndata(count2)
REPEAT
EXITIF temp>=ndata(count1) THEN
ENDEXIT
ndata(count1+1)=ndata(count1)
count1=count1-1
UNTIL count1<low
ndata(count1+1)=temp
NEXT count2
END
```

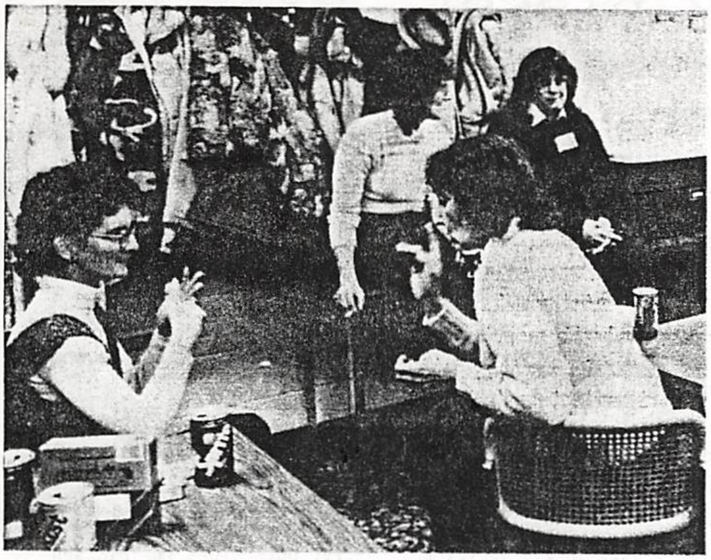
The next sort we will consider is the shell sort. We have already seen that a sort takes less time if the list is already mostly in order. The idea behind the shell sort is just that, to get the list mostly ordered then go back and do a more detailed sorting. It does this by dividing the list approximately in half and then by comparing item 1 to item (list/2) it does a kind of bubble sort. By using such a large gap between compared items, an out of place item can be moved long distances quickly. A second pass is made at the same gap if any exchanges were made. When a pass has finally been made with no exchanges, the gap is closed by about half and the process run again. Eventually the gap will be only one. At this point the sort has become a true bubble sort but the list is almost in order, so the process quickly finishes the sort.

Running a shell sort on the same data used for the bubble sort dramaticly shows off the advantages. On a sort of the ramdom data, it took 36 seconds vs. the 980 for the bubble sort. On the inverted data the shell ran for only

28 seconds. The reason this took less time even on what is supposed to be a worst case is that the data became in order even more quickly because it was already sorted and the large displacement of the swaps was able to very quickly get the data in the proper order. On the partially sorted data, the shell sort turned in a respectible 19 seconds. This is clearly not as fast on a partially sorted list as the insert sort, but its not horrible either and if we had to pick one sort to do all around yoman's work, this is the best so far on this size list.

```
PROCEDURE shellsort
PARAM low,high,ndata(1024):INTEGER
DIM temp,top,d,i,j,k:INTEGER
d=1
WHILE d<=high DO
d=d*2
ENDWHILE
d=INT((d-1)/2)
WHILE d<>0 DO
top=high-d
FOR i=1 TO top
j=i
WHILE j>0 DO
k=j+d
IF ndata(k)<ndata(j) THEN
temp=ndata(j)
ndata(j)=ndata(k)
ndata(k)=temp
j=j-d
ELSE
j=0
ENDIF
ENDWHILE
NEXT i
d=INT((d-1)/2)
ENDWHILE
END
```

The last one we will consider is the quick sort. Now pay attention as this one is similar to the shell sort by getting groups quickly into near sorted condition, but the quick sort takes this a step further. This sort also introduces another type of operation not available to RSDOS basic. This is called recursion. What that means is that the program can call itself from within itself. When it does so, the orignal process does NOT forget its variables, but keeps them so that when the called process terminates it can use the data generated by the called program to operate on the original values of the variables not passed to the called program. Got that? But how can this be when each process uses the same variable names? Each process generates what are called local variables. That is to say that no other process even knows about their existance. Remember, to pass a variable to another process requires that we pass it by name to the new process and the new process must have a variable declared as a param of the same size and type as the calling program's variable to receive the contents of the passed variable. Even at this point the only thing the new process knows is where the value is stored so it can fetch it and store the results. It still has no idea what the variable name is in the calling program! Upon return



Woman talk at the party held in December

"OUR FLOPPY DISKS"

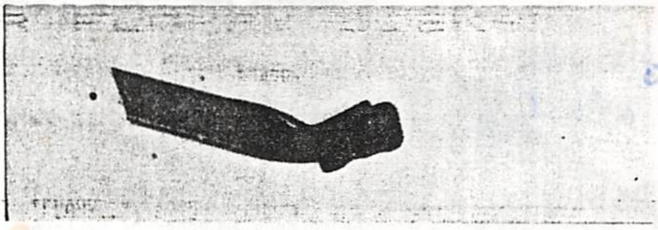
Continued from page 3

a hot-air corn popper, I started to pop corn. At this time the hot air from the popper was blowing all over FLAPPY. Then followed by the hot, dry, itchy pop-corn itself. Soon, FLAPPY was completely covered by a bowl-full of pop-corn. Surely FLAPPY took a real beating this time. Taking FLAPPY out of the bowl, I rinsed it in cold water, wiped it with Kleenex and then inserted in in the jacket. Surely, I thought to myself, that FLAPPY was now a dead disk. But again giving it the same check-up, it still DIR'ed and loaded OK. WOW!!! Just how tough is FLAPPY. How much more abuse can it take?

TEST 5: Having stood up under heat, I decided to test FLAPPY's edurance of cold. Taking FLAPPY out of its jacket, I put it in a Zip-lock bag, half-filled with water, zipped it shut and placed the bad in the freezer compartment of my beer fridge. I left this overnight. The next afternoon, I took the bag out of the fridge, and saw that poor FLAPPY was really stiff in the frozen confines of the ice. It could not even flex its usual self. So I thawed out FLAPPY under the hot water spigot and after FLAPPY was flexible and warm again I wiped it dry with Kleenex and gave it another check-up. DIR'ed both sides, loaded both sides, and amazingly still NO I/O error. What must I do to ruin FLAPPY.

At this point in time I decided that enough was enough. I am retiring FLAPPY so that I can show and exhibit FLAPPY to the world that a floppy disk can survive and was capable to withstanding all kinds of extreme torture and punishment. Also I want to show all that FLAPPY can very easily be converted to a flippy-floppy. Any further testing in the future will be done with another disk.

This article does not intend to encourage others to disregard the usual precautions when holding their disk, and that even thought a floppy disk can actually stand up to such harsh handling, one should refrain from doing so. I just want to point out that if some of the above things happen to a disk, don't push the panic button. Also those who desire can very easily convert their floppies to flippies.



Charles Ross's "re-inker"

MAKE YOUR OWN RE-INKER

by Charles Ross

I recently completed a course in Basic and found that I used my Radio Shack DMPVII printer for several hours a day. This printer did the job for me except that the ribbon had to be replaced three times during the fifteen weeks of the course.

On checking around I found ribbons for quite a few printers cost around $3.50. My ribbons were approximately $9.95 at Radio Shack. I started to analyze my ribbon problem:

1. - Perhaps the ribbons were old and dried out
2. - Perhaps they were too short

Either way the cost could be cut if I could find a way to restore the ribbon. I tried about six different inks without too much luck and finally used an oil-based multigraph ink. That did a good job.

An ink applicator was made by glueing a piece of felt on a piece of metal. Originally I thought I would put the metal in place permanently but found this was not necessar.

The inking process is as follows: The applicator was dipped in the ink and then held against the ribbon on the right side of the printer. A program with short input lines was LLIST(ed) and the as the printer worked, the ribbon absorbed the ink from the felt pad. This inking operation was found repeated three or four times until the ribbon was coated with ink. I found I got best results by doing this as the last operation of the day so the ribbon could absorb the ink for twelve hours or more. The first two or three programs printed smudged the paper the I started to get fairly clean copy. The trick is to coat the ribbon with as thin a coat of ink as possible.

As far as I can see, there is no damage to the printer even after a heavy coat of ink was applied. I estimate the cost of inking including the smudged paper is 1/2 cent per inking OR LESS.

A cheap applicator can be a small brush with a 1/2 inch bristle width.

I intend to use the present ribbon until it shreds.

EDITORS' NOTE: Anyone interested in this inker, see Charles at a meeting and he will explain how to make an applicator.

LIBRARY DONATION

Thanks to Larry Strawn for his contribution of the rompak "MEGABUG" to the Penn-Jersey Color Computer Club Library. See complete listing of what's in the club library elsewhere in this newsletter.

Jan 86
6809 Express

"OUR FLOPPY DISKS"

A real toughy

By Jim Mangan

The following is an unbelievable saga of "FLAPPY," a 5 1/4" floppy disk which was put through a number of tests or trials deemed hazardous to the well-being of disks.

A few days ago I was told by my nephew Tim Foley, who works at Lehigh University Computer Center about some tests they did on floppy disks. The test was to determine how much abuse a floppy disk could take before it crashed. Being of a curious nature, I decided to run my own test. Thereupon I randomly selected from a stack of disks containing many different brands, a disk with the following information on it.

WABASH
Data Tech 1D
Soft Sector
SS/DD
Uninitialized
48TPI 35/40 Tracks
Reorder #: M23A

Let the tests begin!! First I backed up my favorite Jolly Roger disk unto the disk I was going to experiment with. Before beginning I want to point out that the test disk will be referred to as "FLAPPY" from here on in. I tested FLAPPY to make sure that it worked properly. It loaded in a machine language program alright, so now I was ready to start.

Cutting a slit on the edge of the jacket that is held when inserting a disk into a drive, I removed FLAPPY from the jacket using a Kleenex to protect it. I was now looking at FLAPPY which is a brown circular flexible piece of plastic with a large hole in the center and a much smaller hole just off the large one. Before I began the actual tests I had in mind, I re-inserted FLAPPY into the jacket so that the unformatted side could be accessed. I DSKINI(0)'d this side OK and then backed up as I did previously. No checking out both sides by doing DIR's and loading, I discovered that both sides were accessible. EUREKA!!! FLAPPY was now a flippy-floppy without having to cut extra notches and holes in the jacket. The one side was permanently write protected as no notch was ever cut. So now here was a way to back-up a disk unto itself - or how to make a flippy out of a floppy. I think that this was a neat and unique by-product discovery of what I originally intended to do.

TEST 1: Taking FLAPPY out if its jacket, I delibaretly used my fingers. I then rubbed FLAPPY between the palms of my hands. This is supposed to be a NO-NO as instructions on the disk envelope tell you not to even touch FLAPPY. I put FLAPPY back in its jacket and checked both sides for DIR and loading. Results, OK, no I/O errors. See FLAPPY isn't as fragile and sensitive as most everyone thinks.

TEST 2: Again taking FLAPPY out if its jacket, I took it to the bathroom where I washed it with Ivory soap and then rinsed it under the spigot. I then wiped it dry with Kleenex and again tested it out. DIR and Load, OK, No I/O errors. FLAPPY was now the cleanest flippy floppy in the CoCo DEN.

TEST 3: I wrote radially on the bare surfaces of FLAPPY, using a Cross Pen and Pencil set. Using the pen I wrote across FLAPPY from track 0 to 35, and did the same 180 degrees apart and then the same on the other side. Then taking the the pencil I repeated all the above at 90 degrees from where I wrote with the pen. Mind you, this was on the bare plastic surfaces of the actual disk. I did this to make sure that all tracks were written upon at every 90 degrees. I then wiped off the surfaces with a Kleenex and inserted FLAPPY into the jacket and checked it over. STILL good DIR's and loads. NO I/O's. So FLAPPy seems to have a tough skin also, as it seemed resistant to scratching.

TEST 4: Putting FLAPPY into a bowl, which I placed under

Continued on page 4

1986 MEETING AGENDA SET

The Executive Committee of PJ-CCC has set up the following tentative agenda for 1986.

JANUARY	- HOME MANAGEMENT
FEBRUARY	- SPREADSHEET
MARCH	- CARE & FEEDING of your COCO
APRIL	- METHODS to your MODEMS
MAY	- BASIC
JUNE	- HAM & COCO
JULY	- CLUB PICNIC (Sunday, July 27)???
AUGUST	- PRINTERS
SEPTEMBER	- WORD PROCESSING
OCTOBER	- LEARNING is FUN (Education) & Nominations
NOVEMBER	- PASCAL & OS-9, Election of Officers
DECEMBER	- POTPURRI (Anything goes)

The next meeting of the PENN-JERSEY Color Computer Club will be held this FRIDAY, JANUARY 31 at 7 p.m. at NCACC.

President Roni DeGarmo will give a short presention on how to use Radio Shack's "DESKMATE" software for the COCO.

See you there and those who have not renewed their membership for 1986, bring the application which is in this issue of 6809 EXPRESS to the meeting Friday and your dues for $12. Don't forget to fill out the information and make all necessary changes on it if you have gotten more equipment during 1985.

Make your check out for $12.00 to PENN-JERSEY Color Computer Club and give it to Tom Castronuova, new treasurer, at the meeting. You only have to February 1, 1986 to renew and then you will be dropped from the membership list and NO MORE 6809 EXPRESS.

Your dues provides for the printing and mailing of the newsletter and other club functions.

books as source for my material. The first two are "The C Programing Language", by Brian W. Kernighan and Dennis M. Ritchie, both first and second editions. This is the K&R referred to earlier. Mr. Ritchie was the inventor and the first to implement the language. The first edition was for many years the standard to which all "C" language implementations were written. The second edition is an update to reflect the standard created by The American National Standards Institute, hereafter referred to as ANSI C. The C compiler written for the Coco by Microware is written to the old K&R standard. There are a few new "front ends" for the compiler that allows it to accept ANSI style programing. Since these front ends are either PD or shareware, I can supply these as needed. Newer C compilers for other computers are more or less ANSI compatible. We will be using the ANSI standard for the most part to be as portable from computer to computer as possible. Realize that at times you may have to modify some of the code to get it to compile on your machine. Since I cannot know every compiler, I must leave this up to you. I will help where I can and will point possible areas of conflict if I am aware of them.

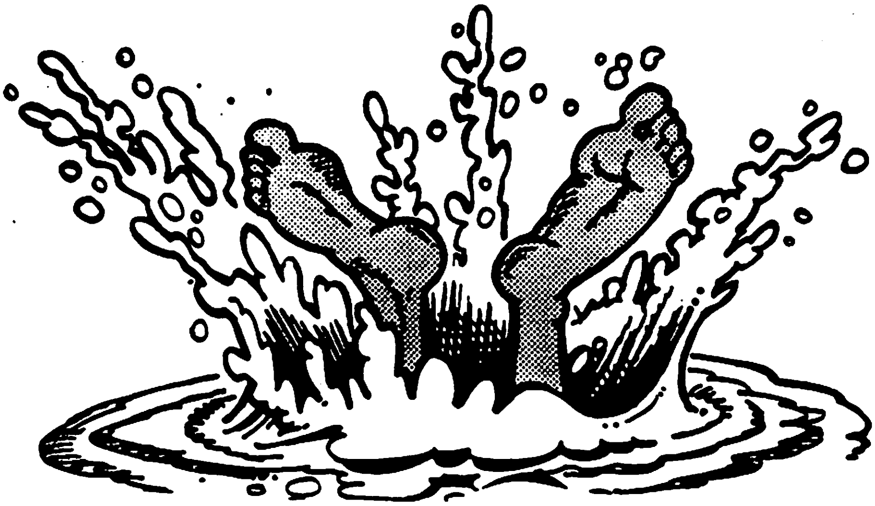
The other three books are "C a Reference Manual" third edition, by Samuel P. Harbison and Guy L. Steele, Jr., published by Prentice Hall; "Teach Yourself C" by Herbert Schildt, published by Osborne McGraw Hill: and the C compiler manual that came with the Microware compiler for the Radio Shack Color Computer. If you endeavor to acquire these books be warned on two of them. The original K&R is out of print and no longer available through normal channels. There are two books out with the title "Teach Yourself C". They are by different authors and different publishers. I cannot say anything about the other book as I have no personal knowledge of its contents. I have read or am in the process of reading these books cover to cover. My opinion is that they are all good references and tutorials with the exception of the compiler manual which states that it was never meant to be a tutorial and directs one immediately to K&R and other books.

Come with me as we explore one of the most universal languages implemented on a wide variety of computers from my Coco to mainframes. Thanks to the ANSI standard, we can now write programs that will work on them all. As wide spread as BASIC is, it was never standardized. As a result, even though BASIC is similar on many computers, it is almost never close enough to run a program written on one computer, say a Coco, on a different machine, say a MacIntosh. But a C program written to ANSI standards can be moved as long as an ANSI compiler is available for the target machine.

For those of you who have the Microware C compiler for the Coco, I have a couple of programs that claim to make the compiler ANSI compatible. I've only tried a few things with these programs installed, but so far I've been able to compile ANSI code successfully.

PAGE 3

ALAN J. WAGNER, SR.



DIVES INTO C !

Here it is the end of another month, my vacation having gone past, and several weekends to boot and I still don't have an article to finish up the program I/we have been working on for almost two years now. I must admit I've lost interest in it. It is a real chore to get myself working on it and when I do, I can't keep my train of thought. If any of the members really want/need the program finished, I will try. If there is no further interest, I will move on to other things that are more in tune with our newly defined and broader base. My apologies to those I may have disappointed, but as I said above, I will complete the program if interest is expressed.

Since the new club arrangement is to embrace as many computer types as possible, continuing to teach only OS9 (even though it is the greatest operating system for any computer and OS-9000 is available for x86 computers) is short sighted. Since I needed to look for something that applies to almost all computers equally, I have decided to discuss the language "C" for awhile. I do not have any grandiose ideas of creating a complete tutorial on "C", but I do wish to expose you to another language other than BASIC. I also will not try to write any large unfinishable programs. Each installment that contains a program will be complete unto itself. OS9 has been accused of being a system of utilities. It will be seen that "C" is a language of small utilities that are strung together to create larger programs.

In our discussions, we will cover the "C" language itself and some of the standard library functions. We will discuss K&R "C" as well as ANSI"C". I will be using five

Basicly Speaking
by Rick Hengeveld

First let me say it was good to hear that some members are actually following along closely with the basic tutorial! It's encouraging to know that the time spent at the keyboards by the writers of the 6809 are being put to good use!

Last Month we started constructing our Database/Cash Register Program. I left ya hanging with an intentional error in line 70, the error had to do with the printing of "Invalid Selection". The corrected line (Printed here) will remove Invalid selection after a few seconds and reprint the menu.

```
70 INPUT A:IF A=1THEN 500 ELSE IF A=2THEN 100 ELSE IF A=3 THEN 200 ELSE IF A=4 THEN 1000 ELSE IF A=5 THEN STOP ELSE LOCATE 5,15:PRINT"INVALID SELECTION!!!":FOR T=1TO1000:NEXT T:GOTO 20
```

```
500 CLS:PRINT"        Data entry Mode"
```

*** This is the screen from where we will enter data***

```
501 GOSUB2000
```

*** Gosub sends the program to the specified line number, in this case 2000. The system will run line 2000 and all other lines after it until it encounters the command "RETURN" at that time the system will come back to the next basic command after the original GOSUB command. In this case it would return to line 504 and the command LOCATE. Remember the next line after the original GOSUB must start with a valid basic command word! If for instance line 504 started with A=100, RSDOS would cough up an error because A=100 is not a basic command***

```
504 LOCATE2,3:LINEINPUT "First Name :";N1$:IF N1$="EXIT" THEN 10 ELSE IF N1$="exit" THEN 10
505 LOCATE2,4:LINEINPUT "Last Name :";NA$
510 LOCATE2,5:LINEINPUT "Address :";AD$
515 LOCATE2,6:LINEINPUT "City/State :";CC$
520 LOCATE 2,7:LINEINPUT "Zip Code :";ZP$
```

Lines 504 to 520 will prompt the operator to fill in the requested information. Name, Address ect. LINEINPUT halts the system while the operator fills in the info. This command allows both text and numerical values to be entered. These entries are stored in the variables you see at the end of the lines such as Last Name being stored as NA$. Notice in line 504 the references to "EXIT". What this will do is to exit this part of the system and return the operator to the main menu.

```
530 LOCATE 6,11:INPUT"Number of Videos";V$:V=VAL(V$)
531 Q=0.00:W=0.00:Q=24.90:W=9.90
535 LOCATE 6,12:INPUT"Number of Audios";A$:A=VAL(A$)
540 T=V*Q:E=A*W:LOCATE6,14:PRINT"Total amount due"
```

```
545 R=T+E:LOCATE
10,15:PRINTUSING"$###.##";R
546 TS=TS+R
550 LOCATE 2,18 :LINEINPUT"Confirm
Data with Customer!  (Y/N)";L$
555 IF L$="Y" THEN 560 ELSE IF L$="y"
THEN 560 ELSEIF L$="N" THEN 556 ELSE
IF L$="n"THEN 556 ELSE 550
556 CLS:LOCATE 5,12:PRINT"OPERATION
ABORTED!!!!":SOUND50,4:FORZ=1TO1000:NEXT:
GOTO500
560 LOCATE 1,20:INPUT"Amount
Tendered";AT
561 CC=AT-R
565
LOCATE9,21:PRINT"Change=";:PRINTUSING
"$##.##";CC
```

***Lines 530 to 565 have to do with the cash register part of the program. ***

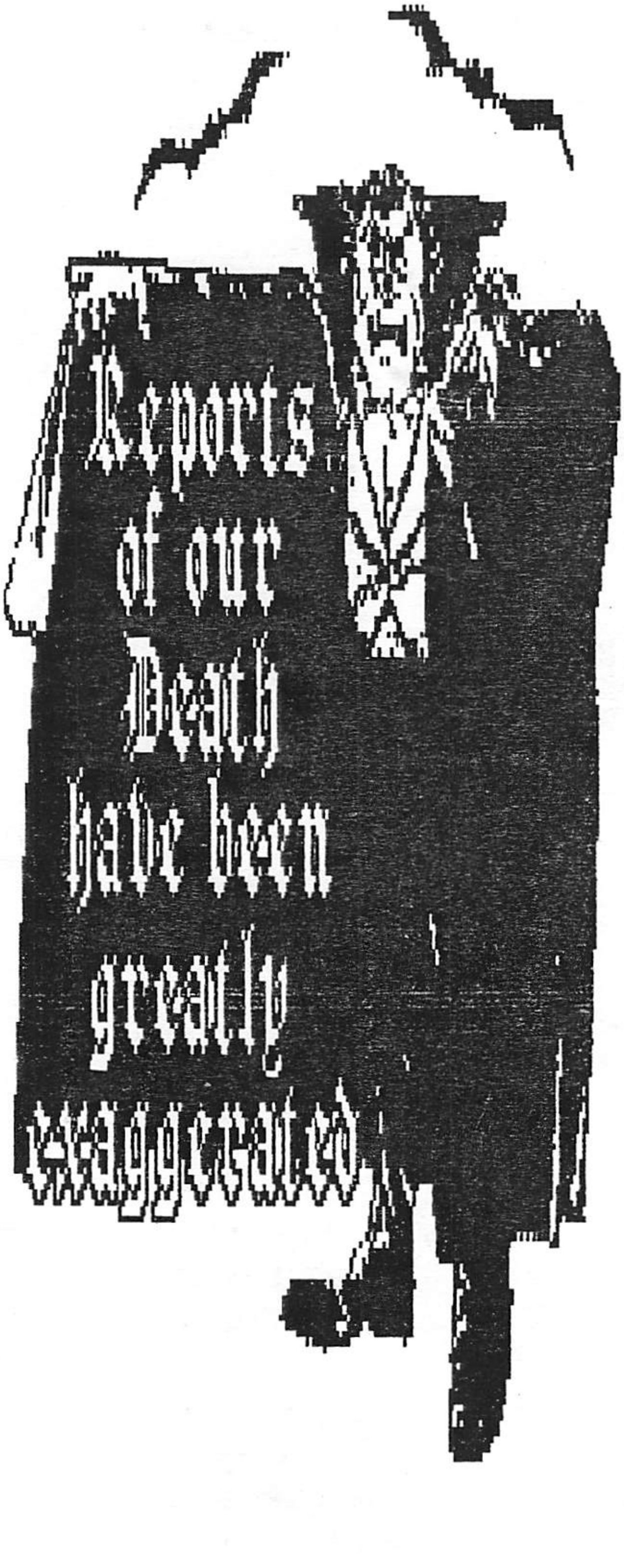
```
2000 OPEN "D",#1, "INFO.DAT",89
2001 RETURN
```

I'll stop here for now, next month we'll take up with the FIELD command, this command is the heart of a database program! I'd suggest you blow the dust of your RSDOS manual and read up a little on the field command.

We're also at a point where you may get a syntax error if you run what you've entered so far, not to worry, we'll tie up these loose ends next time.

****

SEE YOU AT THE MEETING AT 7PM ON THE LAST FRIDAY IN APRIL!



6809 EXPRESS

The Official Publication of The
PENN-JERSEY COLOR COMPUTER CLUB

MAVERICK BBS 215-760-0456

MAY 1991

MM/1 MULTIMEDIA COMPUTER TO BE DEMONSTRATED AT MAY 31 MEETING !

PAUL K. WARD AND MARY KAY WEGLEIN WILL PRESENT NEW COMPUTER

Paul K. Ward, president of Interactive Media Systems, and **Mary Kay Weglein**, IMS marketing director will travel from the Washington,D.C. area to the **PJCCC** meeting to demonstrate the MM/1 computer. The MM/1 operates under OS-9/68000 and is being shipped with a software package which will enable immediate use of the machine.

The MM/1 has graphics, music, and telecommunications capabilities of a very advanced nature. The software package that accompanies the machine includes Network File Manager, PC File Manager that reads and writes IBM PC disks, tape backup support, C compiler, Basic, text editor, two terminal programs, an enhanced graphics editor, dozens of utilities, and more.

A palette of 16.7 million colors is available thanks to the Signetics 664470 graphics chip and the Brooktree palette controller. The MM/1's 68681 serial chip can support a transfer of 500 pages of text per minute with appropriate hardware and software.

Members are encouraged to bring guests to this meeting since the MM/1 will be of interest to a wide variety of computer enthusiasts. The meeting will begin at 7PM. Don't miss it!

THE MAVERICK BBS

Rick Hengeveld-Sysop

Down and Dirty

As of today **The Maverick** has logged over 140 calls! There are over 80 posts in the message base and approximately 40 programs available for download.

Downloading

Since there are many terminal programs available to use with the Coco it would be impossible to cover them all here. Since my terminal program of choice is Ultimaterm I'll talk about how to download using it.

Once you get to the files section of the BBS you may, by selecting "L" from the menu get a listing of the available programs. This listing will pause after it fills your screen (26 programs per screen). Make a note of the EXACT name of any program you would like to download. Pressing "Y" will list the next screen full of programs available. After selecting your programs return to the file menu of the BBS and press "D" for download, Maverick will then prompt you for the name of the program you wish to download. As stated you must type in the exact name, separating the filename from the extension with a period.

(Example) RAMDRIVE.BIN
(UNACCEPTABLE) RAMDRIVE/BIN

Once Maverick has found your file it will then ask what kind of protocol do you want the file sent in? (X) (Y) or ASCII.

I recommend you transfer all files in the X Modem mode as this mode has proven to be totally error free so far. Once this point has been reached Maverick is ready to send the file, you now have about 20 seconds to ready your terminal to receive the file before Maverick decides that there may be a problem and aborts the transfer. These steps are the same no matter what terminal program you use.

After Maverick tells you that its ready to send the selected file you must (If using Ultimaterm) press ALT X , The Alt key first and then the "X" key while still holding ALT down. This will then open a window with file transfer options. Pressing "X" will ready Ultimaterm to receive a file in xmodem. The options window will close and a filename window will open asking the name of the file you are about to receive. Type in the file . name with extension, again separated by a period and a colon with the drive you wish the file to be saved to.

(EXAMPLE) RAMDRIVE.BIN:1
One last thing, Ultimaterm still needs to know what type of file it is saving to disk.

A final window will open up after you've input the saved files name asking what type of file you're saving. The selections will look like this.

(1) ASCII File
(2) Basic file
(3) Basic Binary File
(4) Machine Code file
(5) Text Editor file

Files with the Ext name (REP TXT VIP) Should be saved as ASCII files.
Files with the Ext name (BAS) should be saved as (3) Basic Binary
Files with the EXT name (BIN ARC AR)should be saved as Machine code files.

Options 2 and 5 are rarely used and an explanation of them would take too much space at this time. Ultimaterm will now start to receive your selected file, while the xmodem protocol will check the transfer for any errors, if it finds an error in the transfer it tells Maverick to re-send the damaged part of the file!

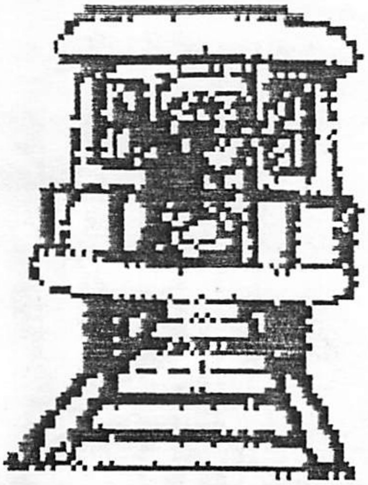
Geez I gotta do all that in 20 seconds or the BBS will abort the transfer?! Actually were only talking 3 keystrokes and a file name, so rest easy!

Ultimaterm is available from the club library. All docs are right on the disk with a handy print program to dump the file. Be forewarned the manual is some 40 pages long! Don't let that scare you it really quite simple to run.

###################
THE MAVERICK BBS
(215) 760-0456
###################

TREASURER'S REPORT
By Clyde Gano

Statement date 4/29/91		
Bal. on hand 3/29/91		$364.42
Receipts-		0.00
Total		$364.42
Disbursements-		
Peter Unks Stmps	Ck#240	$ 7.25
NCCC donation	Ck#241	50.00
Hillcrest Shop		
2 month prntng	Ck#242	19.50
Bell PA April	Ck#243	8.42
Total Disbursed		$ 85.17
Bal. on hand 4/29/91		$279.25



THE LIBRARY CAR

By Al Wagner

Welcome to the May '91 trip of the Library Car. Sorry 'bout the mess. I'm up to my ears in disks and Hardware form the club's library. You'll notice though I cleared a path down the center of the car in anticipation of your visit. Pull up a crate and we'll chat a while.

I usually try to tie the library report to the subject discussed at the meeting to let you know about the material we have in the library that is related. With the discussion being about "other" computers than our own, I can't really tie it in. I've decided to talk about the BBS and what's available in the data base. Looking at the USERLOG on the BBS, I can see that many of you have not yet logged on. I'd like to encourage you to do so for a couple of reasons. There is a growing number of E-mail letters on the BBS. Almost all of them are listed as public. (Of 79 letters only 3 or 4 are private ie., only readable by the SysOp and the party to whom the letter is addressed.) All the letters make interesting reading and you just might get a problem solved if you leave a letter with a question. Rick H. is there all the time he is home and I'm on there usually several times a week if just to check on what new letters have been left there. If you use the "Page The Sysop" option often Rick will reply and you can have a direct conference with him on-line. With our BBS being a single entry port system, you won't be able to have an on-line conference with any one other than the SysOp, but you can leave messages for anyone that uses the BBS.

The database is another good reason. The database is really part of our Library. The nice part about this segment of our library is that you don't have to wait for the meeting to get what you need. Just log on and download it! If you don't see what you need drop Rick or me a note and we'll try to get it to you. If you have created a neat little routine that you find useful, chances are that someone else in the club would find it useful too. You can share it quickly and easily by uploading it to the database. On this BBS that makes it instantly available to everyone else. It should go without saying that only public domain or properly documented shareware should be uploaded.

There are currently (as of this writing) 5 applications, 12 COCO games, 1 MS-DOS game, 3 graphics archives, 2 patches, 2 tutorials, and 11 utilities. Most of the entries are for RSDOS, but at least 3 are for OS9.

Try to get up to the BBS and take a look around. You just might enjoy yourself.

MAVERICK DATABASE

Class	Name	Bytes	Xmdm blks	Description
APP	BIO.BAS	6783	53	COCO does your biorythms
APP	CHORES3.BAS	2937	23	Enhanced ver. of CHORES prgm for COCO 3
APP	CHORES.BAS	2694	22	Fairly divides household chores
APP	WINDOWS.ARC	3267	26	RSDOS windowing program with docs
APP	WRITER.BAS	7764	61	PD Word Processor
GAME	3DTICTAC.BIN	7551	59	A 3 level TicTacToe game
GAME	ASTRO.BIN	6075	55	Space shoot 'em up game
GAME	BERSERK.BIN	7089	56	Berserk Arcade Game
GAME	BLITZ.BIN	9483	75	COCO 2-3 Game
GAME	BLOCHEAD.BIN	6297	50	Game
GAME	GALACTIC.TXT	2694	22	Instructions for playing Galactic Confli
GAME	HOW.BAS	5247	41	COCO reads your mind COCO2 OR 3
GAME	KAHALN.AR	18687	146	OS9 game
GAME	KRON.BAS	17535	137	Tron like game
GAME	MISSLE.BAS	3966	31	Missle command game
GAME	PRISON.ZIP	57471	449	You have to check it out 'SOLITARY' (MS-DOS)
GAME	TREK.BAS	19593	154	Star Trek game for COCO 2 or 3
GAME	VIC.BAS	17343	136	Interesting COCO 2-3 game
GRAPH	GIF.ARC	49641	388	Gif pictures and viewer for RSDOS
GRAPH	MAC2CO.ARC	27003	211	MAC picture viewer and pics
GRAPH	PICS.ARC	27003	211	Mac pictures and viewer
PATCH	BUGS.ARC	10818	85	Repairs some RSDOS bugs on CoCo 3
PATCH	MAX2.ARC	471	4	Patch to run MAX2 without Hires pac
TUTOR	MATH.BAS	2403	19	Math tutor for COCO 3
TUTOR	OS9.ARC	33612	263	OS9 tutorial 3 files included
UTI	AR09.DOC	6270	49	Text file for AR09 arcing program
UTI	AR09.OS9	13440	106	Popular OS9 arcing program
UTI	ARC.BIN	2688	22	CoCo data compression/decompression Prog
UTI	BAS.ARC	15096	118	A mixed bag of goodies in basic
UTI	BUDDY.ARC	9897	78	Basic Programer's utility
UTI	COMM.ARC	15657	123	Communication utility
UTI	DSHIRNK.BIN	2559	20	Squeezes an entire disk DSK ext. files
UTI	DU3.ARC	5625	44	CoCo 3 disk utility - slow but sure

President's Report 5-5-91
by Rick Hengeveld

Most of the time I find it pretty easy to write my club reports. There's always something going on in the PJCCC. However this monthly meeting produced nothing new. **Bruce Navarre's** presentation was very unusual though! This may be the first time in PJCCC history that the presentation had nothing to do with Coco. We looked instead at IBM and Macintosh. There are many people who are looking at, and buying these systems. People that are currently Coco owners included. While at Trenton's computer fair this year I picked up an IBM compatible and I took some ribbing for it. "Oh! Looks like Rick went over to the other side!" Not quite, rest easy. This report comes to you via my trusty Coco. Besides this household can easily support 2 computers! There is not a day that goes by without the switch being turned on, in fact there's usually a line to get on the Coco! This has been true since I bought my old Coco 2 several years ago!

We also intend to have a somewhat non Coco demo coming this month. **Pete Unks** will be bringing in material on the new MM-1 system, although the MM-1 could be considered a cousin to the Coco. The purpose of showing these alien systems is twofold. One, its a good idea to see what's available in the other markets and see how they stack up to the Coco and two, let's face it how many times can we look at word processors!

So if you haven't been to a PJCCC meeting for a while because you thought you've seen it all, now's the time to show your face. I guarantee you'll see something new.

****
****

SEE THE NEW MM/1 !
DON'T MISS THE
MAY 31st MEETING OF
THE PENN-JERSEY
COLOR COMPUTER
CLUB !!!!

****
****

by Roni DeGarmo, Secretary

The April meeting of the Penn-Jersey Color Computer Club was called to order by **Rick Hengeveld**, President, on April 26, 1991 at 7:30 PM. The minutes were approved as printed in the club newsletter. The Treasurer's report was read and approved. Librarian stated there was no news. The Sysop stated the BBS is being used. The number is 760-0456.

Old Business

The Trenton computer show was attended by 5 members who reported on the goodies or the "great deals" just missed by seconds. **Paul Hoffman** a noted COCO programmer was there selling out his stock.

New Business

The club voted to contribute $50 to the Northampton County Community College for their yearly fund drive. Although our membership is decreasing the club felt that we could afford the same amount as last year.

Richard Kravits renewed our lease at the college for another 3 months.

No news on the MM1.

Eric Rhyder is in the states and will be flying in tomorrow to the ABE airport at 11:00 AM. All members are invited to welcome him home.

The meeting was adjourned at 8:07 PM.

Bruce Navarre gave a presentation with an Apple, an IBM PS1 and a laptop.

YES! REWARD YOURSELF BY ATTENDING THE NEXT MEETING OF THE PENN-JERSEY COLOR COMPUTER CLUB AT NORTHAMPTON COUNTY COMMUNITY COLLEGE FRIDAY, MAY 31, 7:00 PM ROOM 21

6809 EXPRESS

The Official Publication of The

PENN-JERSEY COLOR COMPUTER CLUB

6809 EXPRESS

The "6809 EXPRESS" is the official publication of the PENN-JERSEY COLOR COMPUTER CLUB. The club is based in the greater Lehigh Valley of Northeastern Pennsylvania including sections of Northwestern New Jersey. Any non-profit organization may reprint any part of this newsletter provided credit is given. PJCCC will gladly exchange newsletters with any other computer club. Send requests to EDITOR, 6809 EXPRESS, Penn-Jersey Color Computer Club, P.O. Box 6355, Lehigh Valley, PA, 18001 PJCCC assumes no responsibility for errors or omissions. PJCCC assumes no liability for damages resulting from the use of any information or programs contained in this newsletter.

Get on line with

THE MA [redacted]

215-

6809 EXPRESS

The Official Publication of The

PENN-JERSEY COLOR COMPUTER CLUB

H. Peter Unks, Editor

FIRST CLASS MAIL

M/M Clyde Gano
[redacted]