National Speleological Society

Computer Applications Section

CAS Newsletter

Issue #5 December 1981

Welcome to the fifth issue of the CAS Newsletter and the start of its second year of publication. Several apologies are in order. First, the issue is late and it is totally your Chairman's fault with no attempt made to delegate the blame to others. Second, the printing quality of the last issue was terrible and this was due to a poor ribbon in the computer printer where the Newsletter was prepared. Hopefully this issue will be more legible. The third and most important apology is for an error in the address of the CAS Treasurer, Rich Breisch. His address was misprinted not once but THREE times in the issue. Since two of the misprints were on the questionnaire and renewal forms, it is quite possible that some renewals were lost in the confusion, even though Rich notified the Post Office that there might be some misrouted mail for him. Anyone that had their renewal returned to them should send it again to the following address:

Richard L. Breisch
2903 1/2 Upshur St.
San Diego, CA 92106

Cavers (and other people) are notorious for putting bills aside and then forgetting them. The CAS membership renewals have been coming in slowly and we have decided to give everybody another chance to renew before we prune the mailing list. Please check the address label on this issue to see if the number next to your name is 4. This number is the last issue that you will receive. Since this is issue #5, if the number on your address label is #4, you have not renewed and this is the freebie issue. A copy of the membership questionnaire and renewal form are included in this issue for either renewals or for new members.

The results of the membership questionnaire were supposed to be be published in this issue but due to the confusion they will be summarized in the next issue. A current copy of the CAS membership list will also be in the next issue.

CAS SESSION AT THE 1982 NSS CONVENTION

The 1982 NSS Convention will be held in Bend Oregon from June 27 through July 3. There will be two sessions of special interest on Tuesday, June 29. The first will be in the morning and will be devoted to the applications of computers in caving. Anyone (CAS member or not) who is doing anything interesting is encouraged to present a paper, slide show, discussion, or whatever to let others know what is happening. I

(Bob Hoke) am chairing the session and there is no set format or agenda. It will not be as formal as the International Congress and I hope that there will be time for informal discussions following the various presentations. If you are interested in presenting material at the session, please let me know so that I can set up a schedule for the session. We are open to any suggestions for topics of suggestions for speakers.

The second Convention session that will be of interest to CAS members is on Tuesday afternoon and will cover a variety of topics in the management of large cave projects. Since many CAS members are involved in one or more of the large cave projects, this session should be interesting and useful. The session will be chaired by Paul Stevens (5964 Seabright Rd., Springfield VA 22152) and he is interested in any suggestions for topics to be discussed. It is expected that the computer and large cave sessions will complement each other and not cover the same material. Paul suggests that people bring material from their projects that might be discussed, such as maps, histories, training manuals, lead lists, trip reports, etc.

BRITISH COUNTERPART TO CAS

Rane Curl sends word that there seems to be some interest in Britain to the formation of an organization similar to the CAS. Nigel J. Dibben (97 Worth Clough, Poynton, Cheshire) is distributing a questionnaire to anyone interested in caving and computers and he plans to distribute the results to those who respond. Topics of interest include experience with computers, ideas for applications (not necessarily related to cave surveying), sharing of computer facilities, software to share, and specialized hardware developments.

CAVE FILES

One of the most common applications of computers to caving, aside from support of mapping efforts, has been in the automation of various types of "cave files". Normally a cave file is some type of inventory of the caves of a region such as a state or county. These inventories vary in complexity from a simple list of the caves to a quite comprehensive list containing detailed geographical, geological, biological, or similar information about each cave. Many cave files are simple card-image systems but there are several that make use of sophisticated data management systems and allow interactive updates and powerful retrieval capabilities.

The CAS Newsletter is the official publication of the Computer Applications section of the National Speleological Society. Full membership in the CAS (including voting rights) is open to any member of the NSS who is interested in the application of computer technology to the science of Speleology. Associate membership is available to non-NSS members. Annual membership dues are $3.00 and membership includes the CAS Newsletter. Requests for membership applications can be sent to any CAS officer (listed below). Items for publication in the CAS Newsletter should be sent to the Chairman.

Current CAS Officers:
Chairman: Robert Hoke, 8727-12 Hayshed La., Columbia, MD 21045
Vice Chairman: Robert Amundson, 3802 Highwood Ave., Roanoke, VA 24012
Treasurer: Richard Breisch, 2903 1/2 Upshur St., San Diego CA 92106
Secretary: Linda Baker, 3205 Shandwick Pl., Fairfax VA 22031

Unfortunately, cave files are very sensitive since they provide a summary of the information about the caves of a region, often including cave locations and information about formations in the caves. It is no surprise that many people who have developed cave files are very reluctant to distribute them or to discuss them for fear that various types of pressures will be brought to bear to obtain the information in the files.

Keith D. Wheeland of Penn State University (2191 Mt. View Ave., State College, PA 16801) has developed a very comprehensive cave file system that covers the state of Pennsylvania. There are at least 50 data fields associated with each cave and the data is stored on an IBM 4341 computer. He has produced a 50 page user's manual that describes in detail how to set up and maintain the data base. He has offered to wite a paper describing the Pennsylvania Cave Survey and Data Base and we hope to have this paper in the next issue of the CAS Newsletter.

Anyone else who has worked on a cave files system is encouraged to describe it in an article or supply a program description for the Newsletter. In addition to the technical aspects of the system, it would be useful to describe any types of controls placed on the system to prohibit its unauthorized use.

BUYING A PERSONAL COMPUTER (by Peter Lindsley)

Many CAS members are interested in purchasing a personal computer that can be used to process survey data and do other cave-related processing. The following article by Peter Lindsley (5507 Boca Raton, Dallas TX 75230) describes some of the features to look out for in purchasing a computer. The article is reproduced directly from "camera ready" copy that peter produced on his system.

BUYING A PERSONAL COMPUTER?

Are you thinking of buying a personal computer in the next couple of years? Obviously if you ask the guy in the store if now is the time to buy a personal computer his answer will be YES, if you buy one of his. He may be right, but there are some other reasons you should consider if you are interested in buying a personal computer. Unless you just follow someone's advice and buy something "sight unseen", you better consider doing some homework before making up your mind. Personal computers cost as much as an automobile and are probably more difficult to finance, so you should make sure you get the model you need before plunging in over your head.

The choice between a $100. toy and a $6000. 16-bit machine is like the choice between an old Rambler and a new BMW - and you can always get something even better. If you want to spend $100. get the new Sinclair ZX-81 kit. If you want something for $500. look for a used TRS-80 Model 1. If you have an old tape recorder and a TV and $1150., you can get

a new 48K Apple IIC which can be expanded later to a much more powerful system. The same money will get you a TRS-80 Model 3 that is not nearly as expandable but has the screen built in. For $1650. you can get an IBM Personal Computer (16K) that will also plug in to your same old TV and tape recorder. Like the Apple IIC, the IBM will probably be quite expandable in the future as soon as the plug-in boards are introduced. The IBM is a nice machine but it probably costs around $1000. more than the equivalent Apple IIC if you have several accessories on it. If you are a tinkerer at heart, consider a "Big Board" kit from Digital Research Computers (Garland, Tx.) for $500. Add a power supply, keyboard, monitor, a couple of 8 inch drives and make your own box and you have a rather powerful CP/M machine for less than $2500. For $3500. consider the Apple 3 versus the IBM. The Apple 3 has been out for a year longer than the IBM, but early reliability problems (now fixed) with the Apple 3 slowed software development and some consider the choice a toss up between thesetwo machines. If you can afford $5000. and upward, you have an even more difficult choice. You could purchase a fully loaded Apple 3 or IBM, a Fortune 32:16, the new TRS-80 Model 16(said to be available by June '82), or wait to see what the Apple 4, Commodore, or Sage II new 16-bit machines will feature when they are announced. Yet another choice is an S-100 system, well proven in the small business environment during the last few years.

The way you choose between the $100. and $6000. machine, besides the obvious difference in cost, is to carefully determine what type of software you need, the amount of on-line memory you need, and how much mass storage memory you want with relatively fast access times. You need to do your homework because this is where the fast talking salesman will try to snow you with strange new words. This is where you can benefit from visiting your local computer club, as many different ones as you can find, and get some first hand information. Check out all three items mentioned above because once you get a particular machine you are somewhat locked in by it's design. Monthly computer magazines are also a big help because they can give you an idea of the latest technological advances as well as prices. One of my favorite magazines is *BYTE*.

Software should determine your machine choice as much as anything. There is much more junk or bad software than there is good software. Sometimes both are expensive. The largest software base runs on an operating system called CP/M; all true CP/M systems require a Z-80 or 8086/8088 type microprocessor. The early S-100 microcomputers had Z-80s and the business software was developed using CP/M to drive mostly 8 inch floppy disk drives. WORDSTAR, perhaps the best word processor software, runs only on CP/M. dBASE II, perhaps the best relational data base software, runs only on CP/M. You can run CP/M on most of the small personal computers, if

properly modified, including IBM, Apple, Zenith, and some of the TRS-80 models. The second most plentiful group of software is available for the Apple, using their DOS format (folk that appreciate larger computers don't stoop to call it a 'Disk Operating System'). Most of this software is available at low cost compared to other software. A new operating system that is popular in some sectors is the UCSD Pascal system, but only a few people have learned the language and the software is rather sparse at the present time. Except for the Apple, the available machines usually limit you to one or two types of software so make sure you buy the one you need. The Apple is highly adaptable to many different microprocessors and will run almost any software with proper modifications.

The amount of on-line memory is a new area for consideration. Most personal computers sold today implement an 8-bit microprocessor with a 16-bit address bus. This means they can uniquely identify 65,536 memory locations which is the same as "64K bytes" or the number 2 raised to the 16th power. New machines that implement one of the new microprocessors with:

1) 8 or 16-bit data bus
2) address bus of 20, 24, or more bits
3) 16 or 32-bits used for internal operations and
4) populated with the new 64K RAM memory chips

will be powerful indeed. One of the most interesting new processors, the Motorola 68000, has a 24-bit addressing capability which means 16,777,216 locations or 16.7M bytes or 2 raised to the 24th power! Of course, not to be outdone by the 68000 processor, the 8-bit machines have come up with some new tricks called bank switching which allow addressing several banks of memory (but not at the same time). You can buy the Apple 3 with 256K on the main board and can plug in 256K total in the IBM when they start shipping memory boards. The 68000 is the processor to watch for as the new machines are announced in 1982 - consider holding out for one of these machines and buy it with minimum memory. Fill it up next year when the price of 64K RAMs drops.

The third area you should study as part of your homework is to determine how much mass storage you need and how fast you want to talk to it. The $100. Sinclair can store it's programs on magnetic tape as can a large mainframe computer. The problem is speed, not storage cost or room. Tapes take a long time to load on personal computers and they are not always particularly reliable. Once you have experienced the speed and ease of a disk drive you will have to work one into your budget. Unfortunately, they are not cheap! Expect to pay $500. and up for the first floppy disk drive which includes the special controller circuitry that allows the computer to address the data stored on the disk. (One of the disadvantages of the TRS-80 Model 3 machine is that the first

disk drive costs over $800.!) The standard sizes of floppy disks are 5 1/4" and 8", single or double sided, and single or double density. The density determines the amount of data the disk will hold, and there is almost NO standard format/density to allow exchange between machines. The closest thing to a standard format is the single density single sided 8" disk (IBM 3740 format); and you guessed it... the IBM personal computer can't even read this format (yet). A single density 5" disk holds about 90K bytes (TRS-80 Model 1) and a double-double 8" disk holds about 1.2M bytes (TRS-80 Model 16). Two disk drives are nice for copying disks and for many programs that access the disk often. The IBM Pascal language requires two drives for program development. The latest technology in disk drives is the hard disk, usually implemented without removable disks. The new buzz word is Winchester drive (named after one of the early models which offered 30-30 megabytes on a dual disk) and most of the personal computers will be bringing out a Winchester drive in the future. Profile, the Apple 3 5M byte Winchester, sells for around $3500. It will mix files so you don't have to have a separate unit for DOS, CP/M, and UCSD. The wholesale price of a bare Winchester with controller is around $1000. and up (5-10M bytes) and dropping, so expect to pay less next year.

If you can afford a personal computer and start looking at what is available, be forewarned that this interesting field is habit forming. As soon as you look, you may find yourself buying; and that's not the end. There are numerous magazines, books and new software that you may want to purchase, not to mention accessories such as modems and printers. If you just need a little push to rationalize your extravagent purchase, consider the possible tax advantages. If you purchase a new computer for business use in 1981 and after, you may claim a 10% investment tax credit. For 1982 the IRS has defined the depreciation life of a computer as 5 years with zero value at the end of that time. If you use the equipment in a "research" environment the lifetime is reduced to 3 years. The best part is that if you purchase the equipment on or before December 31, you can still depreciate for the whole tax year!

DRAWING LINES ON A DOT-MATRIX PRINTER (by Bob Hoke)

A lot of people are getting personal computers and are purchasing one of the numerous dot matrix printers that are on the market. Many of these printers are capable of operating in a graphics mode where the user's program can cause individual dots to be printed or not printed. There are typically about 60 dots per inch although some printers have more. These printers are capable of producing moderate quality graphics output and they are certainly adequate for most cave mapping projects.

This article describes how to write a subroutine that will draw a line between two points on the plotter page. The technique described here is general in nature and cannot be applied directly to any printer but it should be adaptable to any dot matrix printer that provides the capability to address individual dots on a line. (The technique is not applicable to printers that only provide a small number of "graphics characters" since these printers do not provide sufficient resolution t. print a credible line at an arbitrary angle.)

Dot matrix plotting techniques require that a memory buffer be

used to store the dots that will later be displayed on the printer and this can use a lot of memory. For example, if a printer has a resolution of 60 dots per inch and we want to display a plot page that is 8 x 10 inches, then 8 x 60 x 10 x 60 = 288,000 bits (or 36,000 bytes) will have to be used to store the image of the plot page. While many of the larger personal computers have this much memory, it will not always be practical to have a single large buffer to store the entire plot page. When this is the case, the plot program must be designed to make several passes through the data to be plotted and only select a portion of the page in each pass. For example, we could plot the top one inch of our 8 x 10 page (requiring a buffer of only 28,800 bits or 3,600 bytes) by ignoring all plot points that are not within the 1 inch high by 8 inch wide "window" we are interested in. After this piece of the final page is built up in memory and sent to the printer, we repeat the process for the next 1" x 8" piece of the page, and so on until we have processed (plotted) all 10 pieces of the page. Note that this technique saves memory but requires several passes through the data to be plotted.

Now that the general technique of building up a plot page in memory has been discussed we can look at how a simple line can be "drawn" to connect two points on the plot page. In reality, the technique simply involves turning on the dots that fall on the line that connects the points. In out discussion we will assume that the origin of the plot page is in the lower left corner and that the location of a point is expressed as the number of dots to the right of the origin and the number of dots above it. For example, (60,150) would be the point 2.5 inches above and 1.0 inches to the right of the origin on a page with 60 dots per inch.

We will first describe a subroutine called POINT that will be used to turn on a single dot on the plot page in memory. For example, CALL POINT(60,150) will set the bit for the point at (60,150) to a "1".

The following subroutine (written in bastardized FORTRAN 77) will turn on a point on an 8 x 10 inch (60 dots per inch) plot page.

```
      SUBROUTINE POINT(IX,IY)
c     define our plot page as 480 dots wide by 600 dots high
c     (we will store 8 dots per 8-bit byte)
      INTEGER*1 PAGE(60,600)
c     first, make sure the dot is within the page (ignore attempts
c     to plot points that lie outside of the page)
      IF (IX .LT. 0 .OR. IX .GT. 479) RETURN
      IF (IY .LT. 0 .OR. IY .GT. 599) RETURN
c     we are now ready to turn on the appropriate bit. Assume that the
c     "TURNON" function turns on a single bit in a byte.
c     (bit 1 = leftmost, bit 8 = rightmost)
      IBIT = IX - (IX/8*8) + 1
      PAGE(IX/8+1,IY+1) = TURNON(IBIT)
c     the desired bit in our plot page buffer is now turned on.
      RETURN
```

Now that we have a simple way to turn on a bit, we will write a

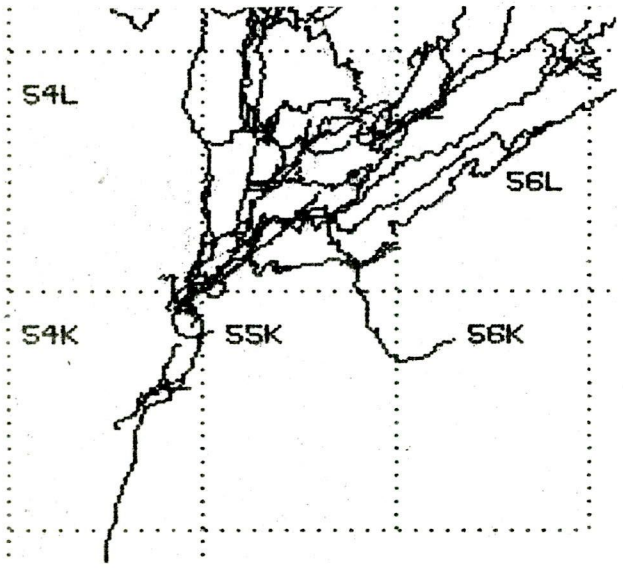
subroutine called LINE that will turn on all of the points between two points using multiple calls to POINT. The parameters to line are the (X,Y) coordinates of the end points of the line. For example, CALL LINE(0,0,60,150) will draw a line from the origin to point (60,150).

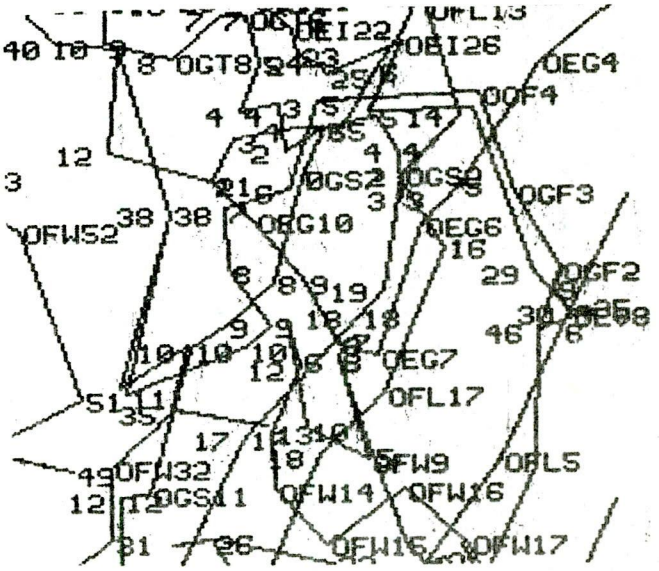
```
      SUBROUTINE LINE(X1,Y1,X2,Y2)
      IMPLICIT INTEGER (A-Z)
      REAL REMAIN, SLOPE
      TX1 = X1                      :: save x-coordinate of starting point
      TY1 = Y1                      :: save y-coordinate of starting point
      DELTAX = X2 - X1              :: compute change in x-direction
      DELTAY = Y2 - Y1              :: compute change in y-direction
      REMAIN = 0.5                  :: start out with 1/2
      XCHG = 1                      :: assume x change is positive
      YCHG = 1                      :: assume Y change is positive
      IF (DELTAX .LT. 0) XCHG = -1  :: if x change is neg, swap sign
      IF (DELTAY .LT. 0) YCHG = -1  :: if Y change is neg, swap sign
      DELTAX = ABS(DELTAX)          :: force positive value of x change
      DELTAY = ABS(DELTAY)          :: force positive value of Y change
c     determine which direction (x or y) is the longest, then look
c     at each dot in that direction and compute the location of the
c     dot in the other direction.
      IF (DELTAX .GT. DELTAY) THEN
         :: x-distance is longer than y-distance
         SLOPE = DELTAY/DELTAX * 1.0
         DO 100 STEPS = 1,DELTAX
            REMAIN = REMAIN + SLOPE
            IF (REMAIN .GT. 1.0) THEN
               TY1 = TY1 + YCHG
               REMAIN = REMAIN - 1.0
            ENDIF
            TX1 = TX1 + XCHG
            CALL POINT(TX1,TY1)
100      CONTINUE
      ELSE
         :: y-distance is longer than x-distance
         SLOPE = DELTAX/DELTAY * 1.0
         DO 200 STEPS = 1,DELTAY
            REMAIN = REMAIN + SLOPE
            IF (REMAIN .GT. 1.0) THEN
               TX1 = TX1 + XCHG
               REMAIN = REMAIN - 1.0
            ENDIF
            TY1 = TY1 + YCHG
            CALL POINT(TX1,TY1)
200      CONTINUE
      ENDIF
      RETURN
```

Once your program has drawn all the lines that are to be plotted, it must call another subroutine, perhaps called PLOT, that will format the plot lines for your printer and actually send them to the printer. The nature of the PLOT subroutine is dependent on your hardware and it

will not be discussed here.

The POINT and LINE subroutines described in this article were recently implemented on a PRINTRONIX dot matrix printer attached to a UNIVAC computer. The original development of POINT and LINE was done by Mike Belch (a non-caver) and I used them to implement a cave plotting program. Two examples of the output from the program are shown below. The left example is part of an 800 foot per inch plot of Organ Cave (WV) and the right example is part of a 50 foot per inch plot of the same cave. Note that straight lines are slightly "jagged" due to the relatively low resolution of the printer in graphics mode (60 dots per inch horizontally, 72 dots per inch vertically). The text in the plots was printed with a SYMBOL subroutine that uses POINT and the dotted lines were done with a DASH subroutine that is similar to LINE except that it draws various types of dashed lines.





PROGRAM DESCRIPTIONS

PROGRAM: NEWASK

LANGUAGE: IBM WATFIV FORTRAN

AUTHOR: George Dasher (55 Kalafat Mobile Village, Buchannon WV 26201

DESCRIPTION: This is a simple, lean, in-quick, out-quick survey program. Input can be either metric or English units and output can be either, regardless of the input units. Multiple surveys are allowed and each can have either a manually entered starting point or can reference a previously entered survey station. The program closes loops sequentially.

The output is listed in table form by individual survey displaying station number, X, Y, and Z coordinates, slope distance, azimuth, inclination, and horizontal and vertical distances. The original data is also printed as a check. Printed totals include slope and horizontal sums for each survey# X, Y, Z, horizontal, slope error and horizontal and slope percent error for each loop# as well as the grand totals of slope and horizontal distances, number of shots, number of surveys, and the average slope percent error at the terminus of the final survey.

HARDWARE: Any computer capable of running WATFIV FORTRAN (or of using CHARACTER mode variables). Any FORTRAN 77 complier should also be able to run this program.

PROGRAM: DOT-MATRIX PLOTTING SUBROUTINES

LANGUAGE: FORTRAN-77

AUTHOR: Bob Hoke (8727 Hayshed La. #12, Columbia, MD 21045)

DESCRIPTION: This package is a series of subroutines that allow an applications program to easily use the graphics capability of a PRINTRONIX model 300 dot-matrix printer. The plot page is defined as a matrix of points in memory and routines are provided to draw a line between two points, draw simple upper case text, to determine whether a portion of the plot page is currently empty, and finally to send the completed plot to the printer.

The following subroutines are provided:

INIT	Initialize a plot page (set it to "empty")
LINE	draw a solid line from one point to another
DASH	draw a dashed or dotted line from one point to another
POINT	turn on (or off) a single point on the plot page
PLOT	send the current plot page from memory to the printer
SYMBOL	display a string of text on the plot page
COUNT	count the number of "on" points in a part of the plot page

This subroutine package is directly usable only if you have the appropriate PRINTRONIX printer but the techniques used are probably useful in developing similar capabilities on other dot matrix devices. The total package is about 9 pages long and includes extensive comments.

HARDWARE: The subroutines should run on any computer that supports the FORTRAN-77 IF-THEN-ELSE capability and CHARACTER variables. The package could be easily converted to any other high level language.

CAS MEMBERSHIP QUESTIONNAIRE

What computer language do you use most often: _____

What computer language do you use next most often: _____

What computer language would you prefer to use: _____

What is the primary type of computer you use: _____

What other types of computers do you have access to: _____

The following list shows a number of caving related applications that were mentioned in the previous survey responses. Please place a 'C' next to any that you are currently using and a 'F' next to any that you realistically plan to use during 1982.

___Management of survey data
___Word processing
___Survey data reduction
___Statistical analysis
___Blunder identification
___Cave development simulation
___Cave map loop closure
___"Home computer" survey processing
___Cave location studies
___Membership & mailing lists
___Typesetting
___"Stick map" plotting
___Hydrologic modeling
___Cave File management
___Automatic data acquisition
___Automatic map plotting
___Programmable calculator
___Interactive graphics
___Other _____
___Other _____

Is there a function that you feel the CAS should be performing but is not? If so, please explain:

_____
_____
_____

Your name: _____

Address: _____

City_____ State/Prov: _____ ZIP: _____

Please send the completed questionnaire to: Richard Breisch, 2903 1/2 Upshur St., San Diego, CA 92106.

National Speleological Society

C O M P U T E R A P P L I C A T I O N S S E C T I O N

Membership Application

Full membership in the CAS (including voting rights) is open to any member of the NSS who is interested in the application of computer technology to the science of Speleology. Associate membership is available to non-NSS members. Annual membership dues are $3.00 and membership includes the CAS Newsletter. The completed application should be sent to Rich Breisch, 2903 1/2 Upshur St., San Diego, CA 92106. Please make checks payable to "NSS CAS".

Is this a renewal? yes no

Type of membership desired: Full Associate

NAME: (Please print) _____

ADDRESS: _____

CITY: _____ STATE/PROV: _____ ZIP: _____

NSS NUMBER (required for full membership) _____