S·P·A·U·G
Stanford
Palo Alto
Users Group

PRinT SCreen

The Newsletter of Stanford/Palo Alto PC Users Group

VOLUME 9 NUMBER 8 SEPTEMBER 1991

C·O·N·T·E·N·T·S

AUGUST MEETING

Last Wednesday: August 28th 7.30pm at Turing Auditorium

GAMES! GAMES!! GAMES!!!

Do your Christmas shopping early. Come to the August meeting and check out the gift possibilities for every member of your computing family.

CLUB NEWS

The President's Piece

Whither Goest Thou?

It appears that some major decisions will have to be made soon. As most of you already know, Stanford has told the Stanford Macintosh Users Group(SMUG) that they can no longer use the university and it's facilities. They've been given the boot. They have located an office for themselves off campus, but as of this date they don't have a facility for their general meetings.

We have been working on ascertaining what our status is now. We have the very same organization as SMUG, at least as far as Stanford is concerned. Still, we have not been contacted by Stanford at all. Our efforts have been to find out as much as possible without arousing any suspicion on Stanford's part.

So, what kinds of decisions need to be made? For starters, do we try to stay with Stanford? Maybe lay low without rocking the boat, hoping they won't notice us? Not a very dynamic option, but a possibility. My feeling here is that the group needs to consider becoming independent. We've been at Stanford for a long time, and, for the most part, the arrangement has been a good one. In some ways, though, it has also been stifling and frustrating. I'm sure everyone has been to at least one meeting where the equipment refused to cooperate. Probably most important is the fact that as a student organization, Stanford has a lot of control over our money. In fact, there are serious questions as to whether or not we'll be able to use or remove our current funds in the event that we have to move off campus.

The other side of this coin is the question of who is available to run a new self-supporting organization. Should it be non-profit or for-profit? Where would we meet? What about all the expensive equipment we need for our presentations? What about insurance? Under Stanford's wing we don't have to think about these issues, nor do we have to spend more of our own limited personal time to make things work. A lot of work and effort goes on behind the scenes at SPAUG to keep it up and running, and there is an understandable reluctance on the part of the current board to add to that load.

We're looking for feedback here. And as always, we're looking for help. Give the club a few hours of your time. Get involved and I guarantee that you'll get more in return.

Are You Game?

Do you realize that there are only 125 days until Christmas? Have you started your shopping yet? Have no fear, for SPAUG is here to help. Being the full service club that we are, this month's presentation will combine the finer features of the Home Shopping Network and the Price Club, all under one roof and at one time. To save you the agony of gift decisions for those hard to shop for computer nerds/hackers/users, we've assembled the finest, most complete display of games, simulations and gadgets ever seen under one roof. You'll see Sim Earth, LucasFilm will be there, Broderbund will dazzle you, and much, much more! So bring the whole family down, there'll be something for everyone!

Hooray For Henry!

Can we talk involvement here!? Henry Hollwedel, on his own and voluntarily, has revamped and revised our long lost Novice SIG. He has come up with a great program for new and not so new members. It will be discussed at the next meeting. Be sure to read the curriculum in this very newsletter. I predict that this will turn out to be one of the hottest tickets of the season, so if you've ever wanted to attend a Novice SIG, plan on being at the first class. Attendance will be limited, so it's first come first serve. Scalping of tickets at the door will not be allowed.

Planning Meeting - August 12

This month we combined the planning meeting with a potluck dinner at the Altman Mansion. As has been said many times before, these meetings are open to everyone. The meeting this month provided some of the best reasons ever for attending-Don's vegie casserole, Bob's salad, Beverly's chicken, Tony's fish, Henry's potato salad, Mrs. Smith's apple crumb pie and the piece de resistance, Lou's mousaka. A good time was eaten by all. And thanks again to Beverly and David Altman for the use of their home, pool and ping-pong table.

Subjects we discussed:

- What to do if/when Stanford asks us to leave.
- How to assure that we use/remove our funds from Stanford's grasp.
- After many efforts to secure equipment through a donation, we agreed to purchase a basic system for the BBS. Bob Bottini, Tony Allen and Henry Hollwedel will make the decision of where to make the actual purchase.
- A suggestion was made that FOG would, for a nominal fee, set up and maintain the BBS for us. This would be a tremendous time savings for Bob, who has spent lots of time trying to keep the beast alive. They also have an extensive library of files to choose from. The matter is under consideration.
- The Sidewalk Fair plans were discussed. There are going to be lots of vendors, maybe we could start a little earlier

Software Reviews

We have two new software releases that we would like members to review.

1. **File Shuttle Xpress** - a high speed File Transfer program that transfers files from one PC, or laptop, to another, using either DOS or Windows.
2. **Immunity** - a Hard Disk Mirroring system which provides fault tolerent computing on systems with two hard disks by continuously duplicating information created on one disk on the other - the perfect back-up. Disks of differing sizes can be used.

Naturally the reviewer of the program gets to keep it. Regardless of whether the review is favorable or not!

Sidewalk Fair

There are going to be lots of vendors this year, with a wide variety of products. Be sure to mark it on your calendar—Sept. 25, 6:00 p.m. This is the same night as our regular meeting, so just plan to be there early to take advantage of the bargains. There will also be space available for members to sell their stuff.

Free Trial

A special User Group offer from Ann Arbor Software of their full featured Textra 6.0 word processor. As PC Magazine said, "It's a terrific $500 program that happens to cost a fraction of that amount." Coupons for a 60 day free trial offer will be at the August meeting. If you like it, it's yours for $95.

THE BBS IN PRINT

BOB BOTTINI

Bob is the BBS Sysop

(415) 321-4497

THE NEW BULLETIN BOARD NUMBER

GREAT NEWS ABOUT COMPLETE RENOVATION OF BBS

On Monday, August 12, 1991, the Planning Committee, by unanimous vote, agreed to a complete overhaul of the BBS. We will be buying a new 386/33 Mhz computer. US Robotics has also made available to the club new Courier HST 9600 Baud modems that list for $1000 at a close out price of $229. We are also moving to new BBS software. We are going to use the famous and highly rated (Editor's Choice, PC Magazine, Sept.10,'91) RBBS-PC, Version 17.3b. This is the most widely used BBS software and it is FREE. Let's discuss some of these items in detail.

MODEM OFFER

The world computer communications is rapidly moving into the faster 9600 to 19,000 & higher baud ranges. Some of the reasons for the change are obvious, one of course is time and another reason being phone line charges. As our BBS also needs a new modem I shopped for prices in the higher baud ranges. The least expensive "reliable" modem I could find at 9600 baud was close to $500. A "reliable" 2400 baud modem was $160. By luck we got a letter dated July 22, 1991, from U.S. Robotics announcing a newly engineered line of modems and advising us that they are closing out their present line to Users Groups such as ourselves. If I may quote the second paragraph of their letter:

"In creating modems compatible with the newest CCITT 14,400bps standard, V.32bis, we have a number of new end of the line 9600 v.32, HST and Dual Standard modems. These are new modems with a full warranty. Rather than put these modems through our standard channel, we are offering them through September 30, 1991 (while supplies last) to user group members for only $229 - $399 Versus original list prices of $995 to $1,295."

The price of $229 for the product they are offering, a 9600 baud modem that can interface with 1200 & 2400 baud modems as well as operate at 14,400 baud with another US Robotics modem, is a great buy. Order forms are included in this months print screen for your use if you need a modem. It may be of interest for you to know that I bought one of the $160 "reliable" modems a little less than one year ago. It's on its way back to the manufacturer for repair. If you are in the market for a modem you are in luck.

A NEW SYSTEM

The RBBS-PC, Version 17.3b, Has been rated in the current issue of "PC Magazine"as Editors' Choice. A brief history of RBBS tells us that it was first released in the summer of 1982 and was the first DOS-based PC BBS software. It ran using a BASIC interpreter that was included with the IBM PC. Though copyrighted, it is a free BBS program. It is easy to install and 327 different configurations are possible, and explained in the manual. Security extends to 65,536 levels and again it is all explained in an excellent manual that should be read by anyone wanting to run a BBS. The program supports CD-Rom drives. Has a fast file-search facility allows a door feature give access to dBASE and Oracle databases. By using a door feature your can have a system of on-line search of databases. Technical support for this product is available through thousands of BBS Systems throughout the country. It is by far the most widely used BBS System.

LOGGING ON

May I at this point address the access to BBS Systems. In general you start with a computer, a modem and a communications software package. The modem can be an internal modem(which from personal experience I find less acceptable.) or an external modem. The modem plugs into one of the com ports of your computer and you use the communications software to dial up a BBS. When contact is made to the host computer by your computer through its modem, a message is flashed on your screen that announces you have made contact. That announcement usual includes the name of the BBS. It will continue by asking you questions like what is your name, your address, city and state. It will complete its questioning of the caller and will give instructions for use of its facilities. The process is simple, it is easily learned and once you become familiar with the process a whole new world opens for you.

SPAUG NOVICE SIG

HENRY HOLLWEDEL

OUTLINE OF CLASS MEETINGS

It is my intention to teach people about DOS and not about specific application programs. The plan is to have 5 monthly, or perhaps bi-weekly, meetings, each meeting from 7:30 to 10:00pm. In the first program I make the assumption that each person knows only how to turn the computer on. Everyone to bring a blank disk and something to take notes on to each meeting.

SESSION 1.

A. How a computer is organized.

1. Keyboard, VDT, memory (RAM) (ROM), printer
   a. Special keys on a keyboard
   b. Ctrl, Alt, Del keys
   c. Use of the "^" character
2. Disk drives
   a. Different types of disks and drives
   b. Capacities of disks
   c. Tracks and sectors
   d. The hard disk ("C" drive)
   e. Do not place disks near magnetic devices, paper clips; or in a very warm environment
   f. Write on disk labels with felt pens only
   g. Capacity of a floppy disk: 112 files in root directory
3. Input and output ports on back of the computer.
4. What is "DOS"? - levels of DOS
5. What are "files" and how are they named?
   a. Try to use meaningful file names
   b. File extensions:
   c. Executable files - .com; .exe; .bat
6. What really happens when you start up your computer (cold boot)?
7. What is a warm boot?

B. How to determine what files (if any) are on a disk

1. DIR /p, /w, filename
   a. Use of "wildcards"
2. Printing your directory i.e. Ctrl-P or Ctrl-PrtScr

C. Discuss basic commands:

1. FORMAT - options, dangers
2. DISKCOPY
3. DISKCOMP
4. TYPE
5. PRINT
6. COPY
7. DELETE
8. RENAME
9. VOLUME

SESSION 2.

A. Review Session 1.

B. Explain function of

1. CONFIG.SYS
2. AUTOEXEC.BAT

C. Hard disk organization (can also apply to floppies).

1. Directories and sub-directories.
   a. The root directory
2. How to use the DOS prompt.
   a. Changing sub-directories (\cd)
3. In your root directory-
   a. Command.com, autoexec.bat, config.sys
   b. Talk about each of the above
4. The FAT table

D. The concept of path

E. Disk management - copy data onto a floppy for backup, and don't clutter up your hard disk.

1. Date and time notations on a file. No files written and dated before 1-1-80.

F. What is the "BIOS" in your computer?

SESSION 3.

A. Review Sessions 1 & 2

B. How to organize your hard disk

1. What to do in general:
   a. Put only programs on hard disk
   b. Except for frequently used files, keep all data on floppies and back them up with another copy.
   c. Give some consideration to what names you give to files so that you can identify them later if you have to - MEANINGFUL FILENAMES
   d. Have other copies of the programs you have on your hard disk
   e. Run CHKDSK on a periodic basis
2. What not to do:
   a. Format the hard disk (or any disk) without knowing the consequences.

C. Discuss backing up the hard disk - options

"Your hard disk will fail. It's only a matter of when."

1. Use of DOS backup program
2. Backup selectively programs and data
3. Use of a program like XTREE as a file manipulator

D. What is ASCII?

E. Begin talking about Edlin

F. Some reference books about DOS

SESSION 4.

A. Review sessions 1, 2 & 3.

B. Talk about file recovery

C. How and why to format a disk

1. Options when formatting a disk

D. More about Edlin

E. Creating batch files in Edlin

SESSION 5.

A. Review sessions 1,2,3 & 4.

B. What to do with a mystery disk

1. You found it in your disk collection, is it useful?
2. How to execute a program on a mystery disk
3. How to find a file with "wild-cards".
4. How to find a file with "dir"

C. TYPE

1. How to read a .BAT file
2. What's a README file?

D. The dangers of viruses

1. How to keep your system clean

E. What is a menu and how does it work?

F. Customizing your system

G. Spreadsheets, databases and wordprocessing

In broad terms what are they and what do they do?

H. Answer questions.

BETTER MAGAZINE RATES

Ziff-Davis has a discount offer on several of their computer magazines for SPAUG members. Call 1-800-777-2547, tell them you want the user group rate and they'll set you up. The following are available:

P.C. Magazine
$24.97/yr
P.C. Computing
$14.97/yr
Computer Shopper
$14.99/yr
P.C. Sources
$12.97/yr
Mac User (?)
$13.50/yr

A BIG WELCOME TO THE FOLLOWING MEMBERS

NEW

James Perry, Belmont

RENEWALS

Jan Altman, Santa Clara
Bob Birss, Mountain View
Bob Bottini, Redwood City
Jack Egbert, Portola Valley
Harry Hughes, Sunnyvale
Samuel Moise, Los Gatos
Jim Moore, Newark
Mike Van Waas, Palo Alto

SPAUG GOES TO TOWN

PAUL STALEY

I have never been a big fan of games for the PC. Most seem too juvenile for my tastes. Plus, I find the idea of a "game" to be limited. It starts at a certain point, heads in a specific direction and ends where it tells you it should. Far more intriguing to me is the idea of interacting with a sophisticated simulation of a real-life situation.

For quite some time now I have been reading glowing reviews of SimCity(SC), from Maxis Software. This program simulates the interactions of various elements of a city, e.g. residents, transportation, finances, industry and commerce. The best part of the whole package is that you, the player, are completely responsible for creating and maintaining the city of your choice. The program has definite rules that it applies to your creation, rules that control traffic on your streets, the rate of population increase (or decrease), when and where disasters strike. Depending upon how you mix the different elements, your city will grow, stabilize or disappear.

Installation of SC was straightforward. I was disappointed to find that the highest screen resolution available is EGA. It's acceptable on the screen, but VGA and SVGA would definitely increase the level of reality. And also, on my screen at least, the display filled only the middle 2/3rd of the available space. It is imperative to have the display as large as possible with SC, since the edit window shows only a small portion of your city at one time. My final gripe concerns the copy-protection method. SC requires that you type in coded info every time you start. Why are developers still doing this? Yes, I understand the need to protect the investment, but surely, after all these years, someone somewhere has come up with a less annoying method.

Now for the fun stuff! When you create a city, SC "terraforms" the land for you to include areas of water, trees and open space. It displays this in two overlapping windows. One shows the entire city area. Obviously there isn't any fine detail here, just an overview. The other window, the edit window, is your work area. It is here that you will begin to put in "zones" of houses, industry and commerce. These have to be connected together by roads and/or rails, then connected to the power grid(which you layout) in order to become useful to your new Sim Citizens. You, the creator, are the mayor of the new burg and are responsible for all things that happen, both good and bad.

All this creating and laying out seems simple enough. The challenging part is that you have to pay for all this creating!. At the beginning of your city you are budgeted a set amount of money. Every time you put in roads, houses, power lines, industry, etc., the price of these improvements is deducted from your budget. The improvements then start to increase your budget by returning taxes on their value, which can fluctuate. So you have a balancing act to maintain. If you then factor in the effects of crime, disasters and voter confidence, all of which you must fix and, of course, pay for in the process, you begin to have a captivating situation. And this is just the beginner's level!

At any point in the creation of your dream city, you can print it out. This can be either as a one page or eight page printout. The eight pager can be assembled into a large (@32x22) poster. It's very impressive and rewarding to see your handiwork displayed this way. Now if it could be done in color!

I was astounded at how easily and completely I became mesmerized by this game. I've had more free time on my hands recently than I want, and I'd feel embarrassed if I had to tell you how much of it was spent in front of my computer trying to squeeze a new nuclear power plant out of my budget while putting out the fires from the latest jet crash. It is very easy to become very attached to your city, and quite concerned when it begins to decline or change in ways that you had not expected. To watch all this happen before your eyes definitely adds to the impact.

I don't know if this makes me a convert to PC games or not, but I've definitely been converted to the SimCity brand of "infotainment". I say thumbs-up on this one!

Sim City retails for $45.95, and the developers, Broderbund Software, will be demonstrating it at the "Games Night" meeting on August 28th

PLANNING MEETING

September 9 (2nd Monday) at 7:30pm

The location

1670 Oak Avenue, Menlo Park

Help us to make decisions about the club. All members are welcome. You don't have to be a club officer to get your views heard.

BATCH FILES 'N MENUS

CUSTOMIZING YOUR DOS SYSTEM

RICK ALTMAN

(In the August PRint SCreen, Rick showed us how to use ANSI.SYS to redefine the keyboard and create macros. Also how to make batch files interactive. Here he concludes with instructions on building a professional, personalized menu.)

4.

INTERACTION

In the last issue we saw how DOS batch files can be given true interactive capability with the use of a public domain program called QUERY. Used in the middle of a batch file, QUERY stops and prompts you for input. Then the batch file branches one of several different ways, depending upon your input.

With QUERY, you can run a batch file many times, yet change the way it behaves each time by answering the query differently. This is the essence of a menu—being provided with choices in midstream—and that is the final destination of this series of articles.

QUERY is terrific for simple batch files that may branch one of a few different ways, but it only accepts valid ASCII input and it requires that you press <Ret> when you're done with your input. That means you can't press a single key to continue the batch file, and you can't press a function key.

Coming to the rescue in this dilemma will be...YOU. If you follow the instructions in Listing One, you will create from scratch a program called WHATKEY.COM. This program, has two important advantages over QUERY: 1) it only waits for one keystroke and acts immediately upon it, not waiting for a <Ret>; and 2) it accepts function keys as input. If you are unable to produce WHATKEY, you can obtain a copy by writing to the address below.

WHATKEY is the answer for a menu system or some other quick-branching batch file, but this speed and flexibility doesn't come easily—you must learn the confusing, backward language of errorlevels.

IF ERRORLEVEL, THEN $%&#@%

WHATKEY works by capturing the most recently pressed key, analyzing it and acting upon it. WHATKEY identifies the key by its scan code or key code and then DOS tests for the code number using its "errorlevel" syntax. Just like with QUERY, you can set up an "If...Then" command.

The tricky part is the rules of the errorlevel game. The scan code for <F1> is 59 but you can't just use the statement:

```
If errorlevel 59 do_something
```

DOS translates this as "If the key code is 59 <italic>or greater<italic> then execute." Code 59 being one of the lower numbers, most keys on your board will pass this test, and the batch file would run incorrectly. Therefore, your tests must eliminate all unwanted key codes, beginning with the highest codes and working down.

For example, say your department regularly uses two different word processors, Microsoft Word and Multimate. You want to write one batch file that will control either program, asking inexperienced users to choose the program they want to run. The batch file could look like this:

```
1.  echo off
2.  cls
3.  :TEST
4.  echo Press F1 for Word and F2 for MultiMate
5.  whatkey
6.     If errorlevel 61 goto REJECT
7.     If errorlevel 60 goto MATE
8.     if errorlevel 59 goto WORD
9.     goto REJECT
10. :WORD
11.    cd\word
12.    word
13.    goto end
14. :MATE
15.    cd\mate
16.    mm
17.    goto end
18. :REJECT
19.    cls
20.    echo Wrong key — try again...
21.    goto TEST
22. :END
```

Line 3 begins the test for a keystroke. First comes the echo statement on line 4 telling the user what to do. Then WHATKEY is run, causing the batch file to stop and wait for a single keystroke. The anticipated key codes are 59 (F1) and 60 (F2) but you must be prepared for incorrect keystrokes to be pressed as well. So the first test, on line 5, is to test for a keystroke equal to or greater than 61. That would be a wrong keystroke, so if that condition is met, the batch file branches to Line 18, "REJECT", issues a simple error message and then loops back to TEST on line 3.

If the batch file survives line 6, it moves to the test on line 7 where it tests for a key code of 60 or higher. Line 6 has already determined that the code is not 61 or above, so the only way the batch file will stop here is if precisely 60 (F2) were pressed. If so, the batch file branches to MATE and runs MultiMate. If not, it's on to line 8, where the same isolation test occurs—if it's 59, go to WORD, otherwise continue to line 9.

If the batch file has made its way to line 9, the operator must have pressed the wrong key, because the tests for 60 and 59 have already taken place. Only key codes below 59 would have survived this far. Therefore, line 9 doesn't bother to test—it just jumps to line 18, issues the error message and then loops back to line 3, where the test begins all over again.

Looking for That One Right Key...

You can use a similar method to isolate a single key code by using two tests in one. In effect, you can ask DOS to check that the key pressed was higher than A but not higher than B. Say you want to write a command that deletes all backup files, but you want the command to warn the user first, the way that DEL *.* does. For that, you could write the following batch file:

```
1. @echo off
2. cls
3. echo    Type Y to delete all backup files
4. echo    or any other key to quit
5. whatkey
6.    if errorlevel 121 if not errorlevel 122 del *.bak
7.    if errorlevel 89 if not errorlevel 90 del *.bak
```

Lines 3 and 4 ask you to verify your intention of deleting all backup files, and line 5 runs WHATKEY. There are two correct answers here—"y" and "Y". A "y" has the key code 121, so line 6 looks for a key code that is at least 121 but not 122. If it finds it, the command to delete backup files is carried out. Line 7 performs the same test, but for key code 89, the "Y". If both tests fail, the batch file simply ends and the deletion is not executed.

LISTING ONE

To create WHATKEY.COM, make sure that DEBUG is in the current directory or in the path, and at a DOS prompt, type the following DEBUG commands (Debug messages are in plain type, your commands are in bold):

```
C:\>debug
-a 100
xxxx:0100 MOV AH,08
xxxx:0102 INT 21
xxxx:0104 CMP AL,00
xxxx:0106 JNZ 010A
xxxx:0108 INT 21
xxxx:010A MOV AH,4C
xxxx:010C INT 21
XXXX:010E <Enter>
-r cx
CX 0000
:E
-N WHATKEY.COM
-W
Writing 000E bytes
-Q
```

The two methods shown here for using errorlevel programming can often be used interchangeably, so for many batch files, you can use whichever method you feel most adept and comfortable with.

5.

Designer Screens

As we near the home stretch, it's time to put a pretty face on all of the automation that we have built so far, the last step before putting our menu in place. It's also the easiest—all we're going to do is use a text editor as an etch-a-sketch.

Following the blueprint of the last few articles, let's say that you have programmed the first six function keys on your computer to be the six tasks that you perform the most. <F1> runs WordPerfect, <F2> runs PageMaker, <F3> runs PC Paintbrush, <F4> starts Lotus 1-2-3, <F5> loads dBase and <F6> formats diskettes in Drive A:.

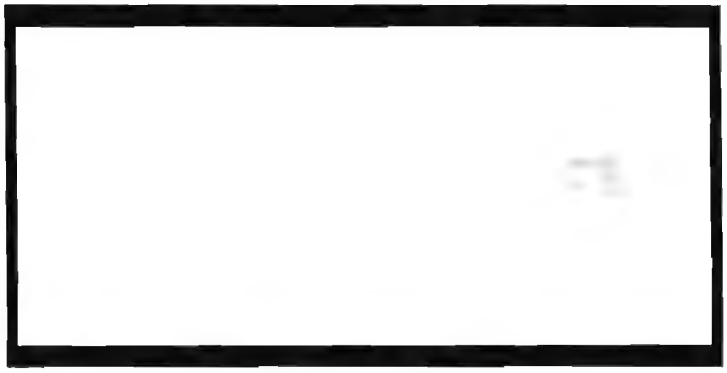
This scheme works so well that you decide everyone in your department should use it, but how do you get everyone accustomed to it without having little cheat sheets lying all over the place?

The answer is to design your own help screen and have it accessible as an executable batch file. While there are dozens of ways to accomplish this using programs available in the public domain, we will look at the one that requires the fewest moving parts. What you will need is a word processor or text editor that lets you type upper-ASCII characters (the line and box characters). This is done by holding <Alt> and typing three-digit codes using your number pad. If you use WordPerfect, save the document as a DOS text file, and if you use Microsoft Word, use the line and box drawing tool and choose UNFORMATTED:NO when you save. WordStar will not work for this task; Sidekick will.

The first step is to enter the text editor and create a simple box. Name the file HELP. The ASCII characters that you will need to type are:

<Alt 220>	**Top of box**
<Alt 221>	**Left side of box**
<Alt 222>	**Right side of box**
<Alt 223>	**Bottom of box**

Now make a box 60 characters wide and 10 lines deep. These box characters are just like normal characters, they just live in a loftier ASCII neighborhood. Use your block copy command to save keystrokes during this task. When you're done, you should see a box like this:



Now comes the easy part: designing your help screen. Making sure that <Ins> is off, now simply type the text as you see fit. If you make typing errors, use the left arrow key and type over the mistake as the backspace key will mess up your box. Also, use the space bar to move, not the <Tab>. Your finished product should resemble Figure 1.

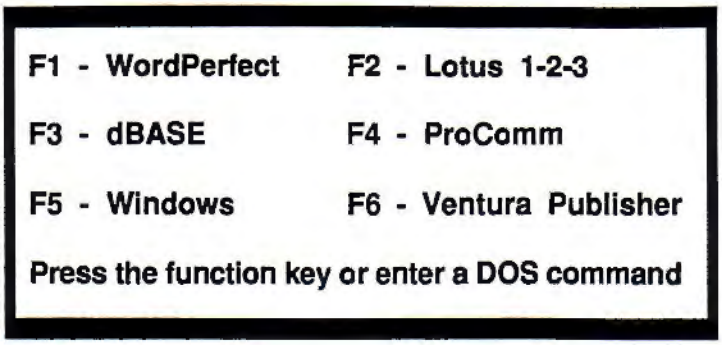


Figure 1

Now create a batch file to quickly show your screen. Call it HELP.BAT and make sure it is written to a snbdirectory that is included in your PATH:

```
1. echo off
2. cls
3. type help
```

Make sure that the TYPE command in line 3 includes the subdirectory in which the help file resides. Notice how your DOS prompt shows up right under the help screen, as if it were one clean unit. This technique assumes a relatively normal DOS prompt—if you have a fancy prompt that puts a message at the top of your screen, you will need to adjust the placement of the help screen.

If the help screen doesn't look just right, use your text editor to spruce it up.

Hooking HELP.BAT into your normal course of affairs is easy—just make HELP the last line of your batch files and the help screen will appear each time you exit any application that was started by a batch file. Also, with ANSI.SYS loaded, you could program a key to instantly show you your help screen. Here is how you would redefine <F10> to be the HELP key:

```
PROMPT $e[0;68;"HELP";13p
```

So far we've covered the following techniques: 1) Defining keyboard macros with ANSI.SYS; 2) Creating batch files that stop and prompt you for input, branching one of several different ways depending upon your input; 3) Creating good-looking screens using high ASCII characters.

Now we are going to put these three elements together to form a menu that will look as good as those on the market today. It will require nothing more than ANSI.SYS and a program called WHATKEY that you will create yourself. In order to cover all the ground needed, I will have to concentrate more on the "what" than the "how".

6.

The Scenario

The menu that we will create here has just six choices, for the sake of expediency. The choices will be made by pressing one of the first six function keys. Yours could use all 10 function keys, and even other keys. The six choices are:

F1: WordPerfect
F2: Lotus 1-2-3
F3: dBASE
F4: ProComm
F5: Windows
F6: Ventura Publisher

We have designed a menu that has a clean, seamless look and allows the user to press just the desired function key, not <Enter>. This screen will be the main menu screen, to which all programs will return upon their exit. Name the file MENU.SCR and test it with the following command: TYPE MENU.SCR. Adjust its vertical or horizontal positioning as desired by placing blank lines above or spaces to the left. When you have it just right, save it and set it aside for now.

Our menu will use WHATKEY (see above) to test if the user pressed any of the function keys F1 through F6. If so, one of those six programs will be executed. Of course, we will have to accommodate for the user pressing wrong keys as well. WHATKEY looks for the "scan codes" sent by each keystroke. The scan codes for the first six function keys are:

F1: 59	**F4: 62**
F2: 60	**F5: 63**
F3: 61	**F6: 64**

Follow the instructions in Listing One to create WHATKEY, and then set it aside.

The Blueprint

Our menu is one large batch file that constantly loops back on itself, until <Esc> is pressed, in which case it ends and returns you to DOS.

The batch file begins by displaying MENU.SCR and then running WHATKEY, causing the batch file to pause. As soon as a keystroke is pressed, the batch file continues, whereupon it tests for the keystroke that was pressed. Depending upon the results of this test, the batch file branches to one of several different labels—either to one of the six programs, or to a simple error message (if a wrong key was pressed), or to the end of the batch file (if <Esc> was pressed).

The six labels that run the programs will have a common structure to them: 1) change directories if necessary, 2) execute, 3) loop back to the beginning of the batch file to start over again. If you are using DOS 3.3, you could call a batch file from within the menu batch file by using the CALL command.

This menu will not actually run any program—you may not have dBASE on your system! Instead it will simply display a message (e.g. "Running dBASE"). Once it is working correctly, you can replace the messages with the actual commands to run each program.

The Batch File

Shown in Listing Two, MENU.BAT is 51 lines long. While that could be intimidating at first, as soon as you break it down into segments, it becomes much friendlier. Here goes:

Line 2 redefines F10 to be the key that will start MENU.BAT every subsequent time after the initial time. Line 3 defines the point to which each program will return when it is done. Line 5 displays MENU.SCR and Line 6 runs WHATKEY.

Lines 7 through 15 provide for every possible keystroke that could be made. With scan code testing, you work from the highest number down, so the first test would be for any scan code above 65, the <F6> key. Any key above 65 would be wrong; therefore, Line 7 would send the batch file back to START. Lines 8-13 are looking for the six function keys, and if any one is found, the batch file jumps to the appropriate label. Line 14 isolates <Esc>, the key that exits the menu, and Line 15 traps the very few keys below <Esc> that can be entered from the keyboard.

Lines 16-20 comprise the first label, the one for <F6>. Lines 17-18 are where the actual commands go, after which Line 19 sends the batch file back to the START label. Some of the lines are indented for readability.

Lines 21-45 make up the other five labels. As you can see, the structure is identical.

Line 46 begins the EXIT label—pressing <Esc> is the only way to send the batch file to this point. Here the screen clears, a simple message appears, the prompt is restored and DOS has control once again. Lines 49-50 are entered by typing ECHO <Alt 255>.

Extras

This is only the starting point, from which you can add to MENU.BAT considerably. Some programs may require a sub-menu, where you can make other choices. To do this, create another screen file that can be typed to the screen, and run a miniaturized version of MENU.BAT inside one of the labels.

You can easily tie in any DOS command (like DIRs, FORMAT, CHKDSK) by assigning them to one of the keystrokes. Perhaps you could call one of the function keys "DOS Commands," and then have that keystroke branch to a sub-menu, where the actual choice of several commands is made.

Using ANSI.SYS commands, you can assign colors and varying intensities to MENU.SCR. Using QUERY, a public domain program discussed in this series, you could have one of your function keys prompt you to run a program that is not on the menu—any program at all. After it is run, the batch file would pick up and loop back to START.

The drawback to MENU.BAT is that it is not as easily changed as the commercial menu programs. To make a change, you would have to edit MENU.BAT and change MENU.SCR. Therefore, if you decide to use it, don't try to make it perfect right away. Expect that the first few weeks will be a process of gradual evolution.

After you have made one menu, you may get the itch to make more. Fear not—they make excellent gifts. They are personal, they look very impressive...and the price is right. Those who add bells and whistles to their menus are encouraged to share them with the rest of us. Have fun.

LISTING TWO — MENU.BAT

(Note: The line numbers are only for reference and are not part of the batch file)

```
 1. echo off
 2. prompt $e[0;68;"MENU";13p
 3. :START
 4.  cls
 5. type MENU.SCR
 6. whatkey
 7. if errorlevel 65 goto START
 8. if errorlevel 64 goto F6
 9. if errorlevel 63 goto F5
10. if errorlevel 62 goto F4
11. if errorlevel 61 goto F3
12. if errorlevel 60 goto F2
13. if errorlevel 59 goto F1
14. if errorlevel 27 if not errorlevel 28 goto EXIT
15. if errorlevel 1 goto START *
16. :F6
17.    cls
18.    echo Running Ventura Publisher
19.    pause
20.    goto START
21.  :F5
22.    cls
23.    echo Running Windows
24.    pause
25.    goto START
26. :F4
27.    cls
28.    echo Running SmartCom
29.    pause
30.    goto START
31. :F3
32.    cls
33.    echo Running dBASE
34.    pause
35.    goto START
36. :F2
37.    cls
38.    echo Running 1-2-3
39.    pause
40.    goto START
41. :F1
42.    cls
43.    echo Running WordPerfect
44.    pause
45.    goto START
46. :EXIT
47.    cls
48.    echo Press F10 to return to Menu
49.    echo *
50.    echo *
51.    Prompt $P$G
```

WINDOWS Q & A

JAN ALTMAN

Q *Why is it when I bring an Excel worksheet over to the Mac that my dates all change? What did I do wrong?*

A **The two platforms have different methods for storing dates.**

Excel stores all dates as serial numbers (the part to the left of the decimal point corresponds to the date, and the part to the right of the decimal point corresponds to the time). When you use Excel under Windows or OS/2, the serial number 1 stands for January 1, 1900; serial number 2 is January 2, 1900, and so on. This is called the 1900 Date System. (It is also used by Lotus 1-2-3.)

The Mac handles dates differently. It uses the 1904 date system. Serial number 1 refers to January 2, 1904; number 2 is January 3, 1904, etc. Both systems allow you to enter dates up to December 31, 2078.

You can select which date system you want to use by the 1904 Date System field in the Options Calculate box. By default, the field is turned off (logical, since you're using Windows Excel). If you're creating a worksheet that will eventually be ported over to Mac Excel, you'll want to adjust the date system. Once you turn the 1904 Date System field on, all *subsequent* dates you enter are stored in the 1904 format. (Realize that previous dates entered are not affected. To avoid mixing both formats on the same worksheet, make sure you set this field before entering any dates at all.)

The change won't affect what you see on your worksheet. You've probably formatted your dates so they look like dates, not serial numbers. But the important thing is, they'll now be interpreted correctly by the Macintosh.

Q *I understand that Word for Windows can do some things a spreadsheet can. I know I can add numbers with Utilities Calculate - is there more?*

A **Word for Windows does have elementary spreadsheet capabilities (note I said *elementary*).**

As you said, you can select some numbers, choose Utilities Calculate, and the result will be put on the clipboard (and appear for a few seconds in the status bar). With this method, you can use + (add), - (subtract), * (multiply), / (divide), % (percentage), or ^ (powers or roots).

Using fields, you can get a little fancier, and they're perfect for calculating in tables. Let's say column 4 of your table has some numbers to add. You've got 6 rows of data, row 7 is blank, and row 8 is for the Grand Total. In row 8 below your column of numbers, enter an expression field as follows:

1) Press Ctrl/F9. Field characters will appear with the cursor inside.
2) Type: **=sum([r1c4:r6c4])**
   (That is, sum the cells from row 1 column 4 to row 6 column 4.)
3) Press F9 to update the field. As long as View Field Codes and Show All are both off, you'll see the result of the calculation. (If one or both are on, you'll see the field code itself.)

Remember that this is *not* a dynamic link. If you change any of the numbers that are being added, you must click inside the field and press F9 to update it. (Printing will also update fields.)

SUM is one of many functions used in expression fields. Check pages 95-96 of the reference manual for a more complete list.

Q *Is there a way to categorize certain days on the Windows Calendar? I'd like to keep separate tabs on things like business meetings, friends' birthdays, holidays, etc.*

A **There's no way to have them appear in separate lists, or search for them separately. But you can give them distinctive marks in monthly view.**

Select the date you want to mark, and choose Options Mark. You have a choice of five different symbols you can use to mark that day. Assign your own meanings to each symbol. You can then tell at a glance when these special days appear each month.

Jan is the Vice-President of SPAUG and a Microsoft Certified Trainer
Send your questions on Windows products to:
3655 Pruneridge Avenue, No. 135, Santa Clara, 95051, (408) 243-5955.

THE SPAUG RESOURCE CENTER

This is a list is of club members who have volunteered their services. If anyone would like their name added to this list, please get in touch with Paul Staley or Jan Altman.

OFFICERS

President	Paul Staley	(415) 493-1582
Vice President	Jan Altman	(408) 243-5955
ASSU Representative	Alex McMillan	(415) 322-4543

MANAGERS

Bulletin Board Sysop	Bob Bottini	(415) 369-2086
Financial Manager	Bev Altman	(415) 329-8252
Librarian - Public Domain Software	Les Weil	(415) 321-5541
Newsletter Editor	Tony Allen	(408) 739-2953

SOFTWARE

Accounting	Larry Mehl	(415) 329-6037
Foxbase	Marie Hooper	(415) 325-1206
Windows Products	Jan Altman	(408) 243-5955
R:Base	Larry Mehl	(415) 326-6037
Lotus 1-2-3	Larry Mehl	(415) 326-6037

LANGUAGES

C	John Watson	(415) 325-7632
Fortran	John Watson	(415) 325-7632
Pascal	John Watson	(415) 325-7632
Smalltalk	John Watson	(415) 325-7632
QuickBasic	Don Baird	(415) 365-6822

CLUB EVENTS IN SEPTEMBER

Sun	Mon	Tue	Wed	Thu	Fri	Sat
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30					

September 9 Second Monday - **PLANNING MEETING**
7:30 PM Beverly Altman, (415) 329-8252 or Paul Staley, (415) 493-1582

September 11 The **WORD FOR WINDOWS SIG** meets the second Wednesday of each month at 7:30pm to talk about Word for Windows and general Windows issues. On Wednesday, September 11, we will discuss:

"MERGING DOCUMENTS & LABELS"

Location is Infotec Training Institute, Techmart, 5201 Great America Parkway, Suite 254, Santa Clara. The group is led by Jan Altman, a Certified Trainer in Word for Windows.

For more information, please call Jan at (408) 243-5955.

There will be no meeting of the **MICROSOFT WORD FOR DOS SIG** this month.

For more information, please call Jan Altman at (408) 243-5955 or Harold Santos at (415) 573-8786.

September 25 Last Wednesday - **Annual Sidewalk Fair & General Meeting**

"SPECIAL GUEST SPEAKER"

7:30 PM Turing Auditorium, Polya Hall, Stanford University

AUGUST MEETING

Last Wednesday: August 28th 7.30pm at Turing Auditorium

GAMES!
GAMES!!
GAMES!!!

Do your Christmas shopping early.

Come to the August meeting and check out the gift possibilities for every member of your computing family.

The Stanford/Palo Alto PC Users Group

P.O. Box 3738
Stanford, CA 94309

Club Information

Meetings	Paul Staley (415) 493-1582
Membership	Beverly Altman (415) 329-8252 $25/year (Students $10)
Bulletin Board	(415) 321-4497
Newsletter	Tony Allen (408) 739-2953

THE AUGUST MEETING

Last Wednesday: August 28th, 7.30pm at Turing Auditorium

GAMES! GAMES!! GAMES!!!

Do your Christmas shopping early.
Come to the August meeting and check out the gift possibilities for every member of your computing family.