

**UNIX Short Course - "Basic through Advanced Unix, FTP and E-mail in Two Lessons"¹
Center for Social and Demographic Analysis (CSDA)**

LESSON ONE

Visit our WWW page for helpful documents: <http://www.albany.edu/csda/>

Contact The University at Albany Help desk if you have trouble: helpdesk@csc.albany.edu - 442-3706

Introduction - UNIX is an operating system. The machine that may be used by all students and faculty at the University at Albany is named "**eve.albany.edu**". This is a SUN computer with a UNIX operating system. UNIX is terse in two ways. First, commands are quite short. They are often abbreviations of their function. The following are examples: "cp" is copy, "mv" is move, and "cd" is change directory. Second, UNIX does not respond when you do something right. It will often let you know if you issue a command that it doesn't like, but if you issue a command that it likes, it will usually just respond with the prompt. Below we will cover several UNIX commands, and we will also demonstrate the use of several software packages that are available on eve - for example, pine for e-mail, pico for editing files, and sas and spss for data analysis.

Connection to a Computer - The first step is to connect to a computer. The machine used by most people at the University at Albany is **eve.albany.edu**. The telnet program you use from a PC must be configured so that it connects to the computer that you want to use. The name must be the full address of the computer. There are other machines on campus, but eve.albany.edu is the machine that must be used for this class. Most other machines on campus have restricted access and may not be used without specific permission. **You must login to eve for this class.**

User ID and Password - You must enter your user id. Your **user id will probably be in lower case**, and so you must enter it in lower case. You must also enter your password exactly, using proper case. **Unlike the IBM or VAX computers, UNIX is case sensitive.**

Terminal Type - The machine will ask you what terminal type you want to use. It will offer you the type of "vt100". If this is the type you want to use (it probably is), just press the "Enter" key.

Command Prompt - After properly entering your user id and password, you should see the command prompt **eve%**. Commands are entered at the command prompt, and submitted to the computer by pressing the "Enter" key.

Home Directory - When you first log in, you begin in your home directory. All users have separate, private home directories. Type and enter the command "**pwd**" to see a description of your home directory. To return to your home directory at any time, from any other directory, type and enter the command "**cd**".

Modifying your environment - Securing Against Unintentional File Deletions - There is a file in your home directory called ".cshrc". It has commands in it that set certain features of your UNIX sessions. We are going to modify that file. We will add an "alias" so that it will be more difficult for you to accidentally delete files. (An alias is a way to modify and create commands. It means, when I say "this", I really mean "that".) We will use an editing software called "pico" to add a file "rmal" to your .cshrc file. Before we modify your .cshrc we are going to copy your .cshrc to another file called cshrcold, in case we make a terrible mistake. The command "rm" (for **remove** file) will be modified.

To make sure you are in your home directory, type:

```
cd
```

To create a copy of your .cshrc file, type:

```
cp .cshrc cshrcold
```

To copy a file to your home directory that you will need in the next step, type:

```
cp /home2/generic/setclass/csda/rmal rmal
```

Now type: "**pico .cshrc**" Move to the bottom of the file by typing **^V** (hold down the Ctrl key, and then press the

¹Created by Patty Glynn and Richard McMillan, Center for Social and Demographic Analysis. 1/25/2000

V or v key) until you are at the bottom. (You don't have to remember these commands. Look at the bottom of your screen, and you will see the pico menu.) Make sure that you are at the bottom of your file. Now, type **^R** to read another file. Pico will ask you the name of the file you want to read. Type: **rmal**. That will bring that file into your .cshrc file. Now exit from pico by typing **^X**. Pico will ask you if you want to save the file. Type **Y** or **y** (for "YES"), then pico will offer the current name of the file you are editing. Hit return, and save the file in that name (.cshrc). Now, in order to invoke the revised .cshrc type "**source .cshrc**". Now test the new .cshrc by typing "**rm rmal**". The UNIX should ask you to confirm that you want to delete the file. If it didn't, something went wrong, and you need to try again or get help.

Directory Structure - UNIX has a hierarchical directory structure that is similar to that on a PC. Each directory level is separated by a front-slash "/", and the very top of the directory, termed the "root" directory, has only a front-slash to identify it. To identify the disk, instead of specifying A: or C: as you would on a PC, you specify the name of the disk, for example /home2.

Exploring and Moving Around Directories - To determine the directory you are in, type and enter the command "**pwd**" (present working directory) at the command prompt. You will see a line of output, beginning with a front-slash. An example is given, but this example will be different than what you see when you enter "**pwd**".
/home1/s/r/rt1234

This example indicates that the user is currently in a directory which is a sub-directory of /home1/s/r, which directory is a subdirectory of /home1/s, which directory is a sub-directory of /home1.

To move to another directory, you can type "**cd**" (change directory), and then the full path name of the directory you want to enter. For example, type "**cd /tmp**" to move to directory /tmp. To confirm that you are indeed in a specific directory, enter the command "**pwd**" (which asks the question, "what is my present working directory?").

Temporary Directory - We will use a temporary directory for the exercises in this course. On eve, the temporary directory is "**/tmp**", and any files or directories created in this directory are erased every night. We will create practice directories and files in this directory so that you will not fill your home directory.

Creating, Deleting, and Listing Files and Directories - The command "**cd**" by itself will always take you to your home directory. Type and enter the command "**cd**" and then type and enter "**pwd**" to confirm your location in your home directory. To list the files and directories contained in your home directory, type and enter the command "**dir**". We will now create a new directory. First, move to the **/tmp** directory by entering "**cd /tmp**". Then type and enter the command "**mkdir**" followed by the name of the sub-directory that you want to create. For example, type and enter the command "**mkdir yourid**" (*substitute your user id*) to create a sub-directory in the /tmp directory with the name that is your id. After creating this directory, type and enter the command "**dir**" to list the files and directories in the /tmp directory. At the far left of the listing you will note that each line begins with either a "d" or a hyphen "-". Directories are prefixed with a "d" and files are prefixed with a hyphen. Locate the directory you created, **yourid**, in the listing. You will note that among the items of information contained on the line representing the sub-directory **yourid** is the date this directory was created. To remove this directory, enter the command: **rmdir yourid**

In order to practice moving from one directory to another, create again (if you just deleted it) in the /tmp directory a sub-directory named "**yourid**", but instead **USER ID**. Use the mkdir command. Type "**cd /tmp**". Check what your present working directory is by typing "**pwd**". Then issue the command "**mkdir yourid**" (*substitute your user id for yourid*). To move down to this directory you have created, enter "**cd yourid**", and then to confirm your location in this directory enter "**pwd**". Make another directory, and name this directory "**one**" (type: "**mkdir one**"). Move down into it using the "**cd one**" command and then confirm your location using the "**pwd**" command. Make another directory in "one/" and name it "**two**" (type: **mkdir two**). To move up one directory, you could enter, "**cd /tmp/yourid**". Or, to move up a directory you could type "**cd ..**" (where the two periods indicate "up one directory"). Try moving up to your directory created in /tmp by entering "**cd ..**" and confirm that you are there with the "**pwd**" command. You should be in "**/tmp/userid**". List the files and directories in the directory by entering "**ls -l**" or "**dir**". You should only see the subdirectory "one" listed. (You can ignore for now what appears to be directories "**./**" and "**../**". These are just indications of permissions that are set for the directory and the directory above.

More information about permissions is presented later.)

```
total 3
 1 drwx----- 3 pjg25   csda      512 Feb 15 15:37 ./
 1 drwxrwxrwt  3 sys      sys       512 Feb 15 15:37 ../
 1 drwx----- 2 pjg25   csda      512 Feb 15 15:37 one/
```

Creating A Text File using the editor called pico - There are a number of text editors for use with UNIX. The easiest to use is an editor named "**pico**." This editor can be invoked from the command line by entering "pico" at the command prompt. If you are not already there, change to the directory you created in /tmp, for example /tmp/*yourid* in our example. Type "**cd /tmp/*yourid***". Confirm you are there with "**pwd**". Then enter "**pico tfile**". You will note that the terminal window changes. Also note that at the bottom of the editor window is a menu of commands, which can be invoked by holding down the control key (designated by "^") and the letter for the command. For our example, enter some text, and then exit and save the file by holding the Ctrl key down and tapping the X key. The editor will ask you if you want to save the file. Tap the Y key to indicate yes. The editor will ask you if you want to save the file in the same name as you typed when you began the editing session. Tap the Y key to indicate yes. You will then see the "eve%" command prompt. Type "**dir**" and you will see the file you have just created listed in the file directory.

Copying a File - A file can be copied using the "**cp**" command. (**DANGER!** The cp command can over-write files without warning.) A file can be copied to another location, to a different file name, or both. Copy the file you just created to a new name in the current directory. Enter the command "**cp tfile textfile**". The syntax of the command is "copy from to". List the files using the "**dir**" command to make sure the file was copied properly. To copy the file to another location, use the "cp" command followed by the filename to be copied followed by the directory path where you want the copied file to be placed. For example, to copy "tfile" to the directory named "one" we created earlier, first make sure you are in the directory where "tfile" is located by using the "**dir**" command. Then use the "**pwd**" command to confirm your location. You should be in the directory you created in /tmp with your id or login name. Then enter the command "**cp tfile one/**". The file tfile will be copied to the subdirectory "one" with the filename "tfile". Note that you could also enter "**cp tfile /tmp/*yourid*/one/**", using the entire directory path *but using your user id or login* to designate the location of the copied file. To copy the file to subdirectory two and also give it a new name, enter "**cp tfile one/two/textfile**" and the file tfile will be copied to the "two" subdirectory with the new filename "textfile". Now change to the "two" directory by entering "**cd one/two/**" and enter the "**dir**" command to make sure the file has been copied to the "two" directory. To copy the file to the next highest directory, enter the command "**cp textfile ../**" and the file "textfile" will be copied with the same name to the higher level directory "one".

Moving a File - The "mv" command can be used to move a file to a different directory, or to rename a file. (**DANGER!** The mv command can over-write files without warning.) Type "**cd /tmp/*yourid***". Type "**dir**" and you should see a file named "tfile". Rename the file "tfile" by entering "**mv tfile txfile**". Enter the "**dir**" command to confirm the file has a new filename. Now move "txfile" to directory "two" by entering "**mv txfile one/two/**" and then change to directory "two" by entering "**cd one/two/**". Confirm that you are presently in directory "two" by use of the "**pwd**" command. Confirm that the file has been moved by typing "**dir**".

Removing a File - The command "**rm**" can be used to **remove** files. (**DANGER!** If you modified your .cshrc as shown on page one, you will have the opportunity to change you mind about removing a file. Without that safeguard, the rm command is very dangerous. Even with it, it is possible to remove files that cannot be retrieved.) We will now remove all of the files we have created. First, if you are not already there, move to the "two" directory. Confirm that you are in the directory "two" with "**pwd**", then list the files typing "**dir**". To remove the file called "txfile", enter the command "**rm txfile**". Because of the precautionary alias you put into your .cshrc file at the beginning of this lesson, the computer will ask you to confirm that you really want to delete each file. You can use the command "**rm ***" (**USE WITH CAUTION!**) to remove all of the files in the directory. The asterisk is a "wildcard". Move up to the directory named "one" by entering "**cd ..**", then remove all files located in that directory (**rm ***). Finally, move up to your personal temporary directory (/tmp/*yourid*), by entering "**cd ..**". Remove all of the files in this directory.

Removing a Directory - To remove a directory, first all files in the directory must be deleted. Go to the "one" directory. Again, confirm your location using "**pwd**" and also the number and name of files located there using the "**dir**" command. Now remove the empty directory "two" by entering "**rmdir two**". Use the "**dir**" command to confirm that directory

"one" no longer contains the subdirectory "two".

Disk quotas - By default, students are currently limited to three megabytes of storage in their *home* directories. Enter the command "**quota -v**" to check how much space you are using. You will receive output similar to the following:

```
Disk quotas for rm3370 (uid 2071):
Filesystem      usage  quota  limit  timeleft  files  quota  limit  timeleft
/home1          3045   3000   3050   7.0 days   441   3000   3050
```

Note that this user is using more space than is allowed. The user is 45 kilobytes over the limit of 3000 kilobytes (3 megabytes), and that 7 days remain to reduce the amount of space taken by files in the directory, or the university will reduce it. **Failure to remain within your quota can result in the loss of important files. Check often!**

Viewing Files -The "**more**" command can be used to view a file. Type "**cd**" to go to your home directory. Type "**more .cshrc**" and the file .cshrc will scroll on the screen. If you have seen enough, entering "**q**" will stop the scrolling. The command "**zmore**" can be used to view a compressed file. (We will talk about these in lesson two.)

FTP (File Transfer Protocol) - FTP, or File Transfer Protocol, is a special *protocol*, or essentially a set of rules and commands, used to both send and receive binary or text files across the Internet. There are several ways to ftp (or send/receive) files. One way is to issue commands at the command prompt of your UNIX account. This will allow the receiving or sending of files to or from your UNIX account and the associated disk storage space. The other common way is to use a windows program, such as ws_ftp, to receive or send files to or from your PC and the associated disk drive(s) attached to your PC. These methods, though both using the same protocol, are very different.

FTP to/from your UNIX account - To access an ftp site, one must first issue a command to both start the ftp protocol program on UNIX and also direct it to a location on the internet of the computer from/to which you want to receive/send files. For example, Microsoft Corporation maintains an ftp site where a number of files are located which are of assistance to users of Microsoft products such as Windows 95, Word, Excel, etc. To go to that site, enter ftp ftp.microsoft.com at the command prompt. You should see something like this. Commands issued are in bold. Start by moving to the directory you created on /tmp.

```
cd /tmp/yourid
eve% ftp ftp.microsoft.com
Connected to ftp.microsoft.com.
220 ftp Microsoft FTP Service (Version 3.0).
Name (ftp.microsoft.com:yourid): anonymous
```

Enter after the colon the word anonymous, which means that you want to connect to the ftp server at Microsoft not as a registered user with an account on the server but simply as a member of the public desiring access to the files. After entering anonymous, you should see the following:

```
331 Anonymous access allowed, send identity (e-mail name) as password.
Password: yourid@csc.albany.edu
```

Following the colon after "Password" enter your full email address. For example: yourid@csc.albany.edu

```
230-This is FTP.MICROSOFT.COM
230-Please see the dirmap.txt file for
230-more information. An alternate
230-location for Windows NT Service
230-Packs is located at:
230-ftp://198.105.232.37/fixes/
230 Anonymous user logged in.
ftp>
```

You will note that now, instead of the eve% prompt, you now have an ftp> prompt. This indicates that you must now use the ftp command set instead of the UNIX commands. However, many UNIX and FTP commands are the same. For

example, enter dir at the ftp> prompt. Following is an abbreviated version of what you will see.

```
ftp> dir
200 PORT command successful.
150 Opening ASCII mode data connection for /bin/ls.
d-----      1 owner      group                0 Mar 13 22:09 bussys
-----      1 owner      group                4405 Nov  5 1997 dirmap.txt
-----      1 owner      group                712 Aug 25 1994 disclaimer.txt
226 Transfer complete.
1254 bytes received in 0.045 seconds (27 Kbytes/s)
ftp>
```

Directories are listed at the left with a “d” preceding them, and that files have no “d” preceding them. We will now download a file. But first a note on file types.

Text v. Binary File Types - A very important distinction exists between text (ASCII) files and binary files. Text files are ASCII text files which you can read with a text editor such as pico. Binary files, such as WORD documents and many executable programs, cannot be viewed with standard text editors and contain characters in addition to the standard set of ASCII characters. To download binary files, you must first set the transmission mode to binary by entering “**binary**” at the ftp prompt. The default for FTP is ASCII, but to make sure you are in ASCII mode, you may type “**ascii**” (lower case). *Binary files will be corrupted and unusable if they are transported in ascii mode.*

Getting a Text File - To get a text file, enter the command “get” followed by the file name. The command “mget” can be used to get multiple files. For our example, enter “**get dirmap.txt**”. You should receive a message on your screen as follows:

```
ftp> get dirmap.txt
200 PORT command successful.
150 Opening ASCII mode data connection for dirmap.txt(4405 bytes).
226 Transfer complete.
local: dirmap.txt remote: dirmap.txt
4405 bytes received in 0.29 seconds (15 Kbytes/s)
ftp>
```

Issuing UNIX commands from within an FTP session - It is possible to execute commands to your UNIX session by using an exclamation mark (!). For example, you can type “**!more dirmap.txt**” now to view the file dirmap.txt. You can type “**!**” to go to UNIX, and then type “**exit**” to return to FTP.

Changing Directories in FTP - Type “**dir**” to see what directories are available. Choose one, and then move into it by typing “**cd dirname**” (substitute the name of the directory you want to move into for “dirname”). Type “**dir**” to view what is in the directory. To move back up the directory tree, just as in standard UNIX you enter “**cd ..**” to move up each level. Now enter pwd to determine which directory you are located within. To change the *local* directory you are in, that is, to change for example from your home directory to /tmp, enter “**lcd /tmp**”. Simply enter lcd to determine which local (on your UNIX machine) directory in which you are located.

Ending the FTP Session - Enter the word **bye** or **quit** at the ftp> prompt to end your ftp session. You will be returned to the UNIX prompt of your local machine. You can now view the ASCII text file (so indicated by the suffix .txt) dirmap.txt by using the pico or the “more” command.

This is the end of the “hands-on” part of lesson one. When you are ready to leave, type “logout” or “logo” or “exit”. You should see a message that you have logged out.

~~~~~  
**PLEASE NOTE: The remaining part of lesson 1 is for reading - it is not “hands-on” exercises.**

**Sending a File via FTP** - You may also use FTP to send files to another computer, but only in directories and computers

where your computer ID has permission to do so. For example, if you have a VAX account, you can use ftp to transfer files between your UNIX and VAX accounts, or between different UNIX machines on campus. You would need to know the full address of the computer you want to connect to with ftp. For example, you could access eve with ftp by typing: **ftp eve.albany.edu** You would be asked for your USERID and your password. At that point, you would see the following prompt: "ftp>" You can use the command "**cd**" to change to the directory where you want to put the file. You can use the command "**lcd**" (local change directory) to change to a different directory on the machine from which you issued the original ftp command. ASCII is the default for transferring files. If you wanted to transfer a binary file (such as a Word or WordPerfect document) you would type "**binary**" and then type "**put myfile.doc**", where "myfile.doc" is the name you want to transfer. The command "**mput**" can be used to send multiple files.

**Printing with Unix** - To print a file, make sure your "pwd" is the directory with the file you want to print. Then issue the command: **lp -c -dlp filetoprint** (Substitute file you want to print for *filetoprint*.) The command to print is "lp". Output can be picked up at LC4 about an hour after the request to print (though the time will vary). The output will be identifiable because your UNIX user ID will be on it.

**Concatenating Files** -To concatenate two files, and create a third file containing the concatenated files -  
**cat textfile1 textfile2 > concatenatedtextfiles**

**Searching for Text in a File** -To search for a text string in a file or list of files, use the "grep" command as follows:  
**grep stratification textf1 textf2**

**Finding a File** - You may use the "find" command to search for a files. For example, if you wanted to search the directory you are in and all subdirectories for files with the extension of .spss, you would type:  
**find . -name "\*.spss" -print** The "find" command has a number of options, so it may be helpful to you to view the manual pages for the "find" command.

**Using Manual Pages (Online Help)** - To view manual or help pages for any command, use the "man" command followed by the command in question. For example, to see the options available for use with the "find" command, enter "**man find**".

**Fixing the Backspace Key** - If your backspace key isn't working, a quick fix is to enter:  
**stty erase <Press the backspace key>** (Enter: stty erase <space key> <backspace key>)  
and your backspace key should work correctly. This will fix your backspace key for the current session.

**Aliases** - As you are now aware, there are many different UNIX commands, some of which can be cumbersome to type each time you want to perform an action. You may find that there are certain commands that you want to shorten. For example, if you must frequently change to the directory /tmp, you can shorten the command "cd /tmp" to "vt" by issuing the command: **alias vt 'cd /tmp'** You can put an alias in your .cshrc so that it will be created each time you log on to your account. Before modifying your .cshrc, it is a good idea to create a copy of it, in case you make a mistake. You may do so with the command: **cp .cshrc cshrc-backup**

## LESSON TWO

**Compressed Files** - There are two utilities available on UNIX to compress files. These are called “compress” and “gzip”. A file compressed with the utility “compress” will have an extender of “.Z”. For example, a file called “raw.data” would become “raw.data.Z” when compressed with “compress”. And, a file compressed with the utility “gzip” will have the extender of “.gz”. A file called “raw.data” would become “raw.data.gz” when compressed with “gzip”.

A compressed file is a binary file. This is important to know if you are using FTP to transport it. You can view compressed files with the “zmore” command. For example, issuing the command “zmore raw.data.gz” or “zmore raw.data.Z” would show the file on the screen. Compressed files must be uncompressed before they can be edited.

We have put several files in a directory called “/home2/generic/setclass/csda”. For this exercise we will work in temporary space. The files on /tmp are deleted periodically. Issue the following commands, but, substitute your user id for YOURID. Make sure that you are logged onto “eve”.

```
cd /tmp
mkdir YOURID
```

Now issue the command (substitute your userid for YOURID): (cp means copy, -r means “recursive” so all files in subdirectories will be copied).

```
cp -r /home2/generic/setclass/csda/ /tmp/YOURID/
```

This will create a directory called “csda” in your directory, and it will contain all of the files /home2/generic/setclass/csda/.

Change into your new directory with the commands:

```
cd /tmp/YOURID/csda
dir
```

The directory should look something like:

```
total 64
 1 drwx----- 2 p jg25 csda 512 Feb 15 16:03 ./
 1 drwx----- 3 p jg25 csda 512 Feb 15 16:03 ../
54 -rw----- 1 p jg25 csda 54596 Feb 15 16:03 exer1.spss.gz
 2 -rw----- 1 p jg25 csda 1886 Feb 15 16:03 exer2.sas.gz
 3 -rw----- 1 p jg25 csda 2066 Feb 15 16:03 exer2.spss.gz
 1 -rw----- 1 p jg25 csda 126 Feb 15 16:03 exer3.spss.Z
 1 -rw----- 1 p jg25 csda 18 Feb 15 16:03 rmal
 1 -rwx----- 1 p jg25 csda 38 Feb 15 16:03 sps*
```

The file named exer3.spss.Z was compressed with the utility “compress” and the file “exer1.spss.gz” was compressed with the utility “gzip”. Either file can be viewed with the “zmore” command. View exer2.spss.gz file by issuing the command: **zmore exer2.spss.gz**

Look at the line that is about 10 from the top that reads:

```
file handle indd1 / inpipe ='zcat /icpsr1/6635/da6635.gz' LRECL=1629.
```

This is identifying a data file that has been compressed using the gzip utility.

Now uncompress the files by issuing the commands:

```
gunzip *gz &
uncompress *Z &
dir
```

The directory should now look like:

```
792 -rw----- 1 pjg25 csda 796250 Sep 14 15:21 exer1.spss
 9 -rw----- 1 pjg25 csda 7209 Sep 14 15:21 exer2.sas
 9 -rw----- 1 pjg25 csda 8951 Sep 14 15:21 exer2.spss
 1 -rw----- 1 pjg25 csda 104 Sep 14 15:21 exer3.spss
 1 -rw----- 1 pjg25 csda 18 Sep 14 15:21 rmal
 1 -rwx----- 1 pjg25 csda 37 Sep 14 15:21 sps*
```

## RUNNING SPSS AND SAS JOBS AND USING THE EDITOR "PICO"

(Refer to <http://www.albany.edu/csda/> for help documents pertaining to SPSS.) The file `exer1.spss` is an SPSS job. The file "sps" is the line of code that makes it easier to use spss. Let's look at the file "sps". Type `more sps`. You will see the following.

```
spss -m -s 10m "$1".spss > "$1".lis &
```

Type `dir`, and look at the permissions on the file called "sps". It should look like: `-rwx-----`. The "x" in the 4<sup>th</sup> column means that it is a file that is executable by the owner (you). This file is a short "script" file. Script files can be used to automate commands. This program will make it easier for you to run spss programs. At the prompt, you can type the name of the file and the code in it will be executed. Before you leave class, you may want to copy this file into for future use.

**Using the Pico Editor-** Now, let's look at "exer1.spss" using the editor "pico". Type: `pico exer1.spss`. We will practice edit commands, but not save the changes. First of all, note that there is a menu of commands at the bottom of your pico editor screen displayed in the telnet window on your computer. Note that to execute a command in the pico editor you must hold down the "Ctrl" key while also pressing the alphabetical key needed to execute the command. The need to concurrently press the "Ctrl" key is indicated by the "hat" symbol (^). Scroll down the file by using either the cursor keys, or by typing `^V` for the Next page. Now go back to the top by hitting `^Y` for the Previous page until you are at the top. Go down two lines and type `^K` which is the Cut Text command. This will cut a line. `^U` which is the Uncut text command will uncut the same text. By cutting a line, and then moving to another place, you can cut and paste text. Type `^G` to Get help and briefly review the additional help in that menu. Scroll through the help menu and note that there are additional commands offered by pico that are not listed at the bottom of the pico editor screen.

**Cutting and Pasting Blocks of Text** - One task that I found difficult to figure out from the help is how to cut and paste blocks of text. Note the following in the extend help menu (under `^G`): "`^^ Mark cursor position at beginning of selected text.`" This means to hold down the Ctrl key, hold down the shift key, and then press the 6. The first ^ (hat) indicates the "Ctrl" key and the second ^ is obtained by pressing Shift+6. Exit from help (`^X`). Now, choose some text at the top to cut and paste. Type `^^`, and then move the cursor. You will see that text is highlighted. Type `^K` and the text will be cut. Move to another place, and type `^U`, and the cut text will be pasted. Now, exit from the file (`^X`). Pico will ask you if you want to save the file. Type "N" or "n" -- indicating that you do not want to save the file. Now, let's run this job. Type the command: `sps exer1 &`

The "&" asks the UNIX to run the job in background. It allows you to do other things while the job runs. Of course, because the UNIX is multi-user environment, you can also login again, and have two (or more) windows open at the same time.

You will see the message "Broken Pipe". This is because the SPSS program is reading compressed data. It is normal to see these messages - there is no need to run home and check your plumbing.

A way to check on the status of jobs is to type the command: `ps`. This gives you a list of all of the jobs your are running on the machine you are using. Also try typing the command "`w`". This shows who is logged on, and what they are doing but it will not give you job numbers.

When your job has finished, you will see something like the following:

```
End of job: 10085 command lines 0 errors 0 warnings 37 CPU seconds
```

Now look at the files in your directory. You will have a new file called “`exer1.lis`”. Issue the command “`pico exer1.lis`”. Browse the file. Go to the very end of the file with `^W` (for Where is) and then `^V` (for Lastline). When you are done, exit (`^X`).

The program `exer2.spss` reads data, and saves an SPSS systems file. Run it now with the command:

```
sps exer2 &
```

You will get a “Broken Pipe” message, but ignore it. When it has finished, you will see a line something like:

```
End of job: 234 command lines 0 errors 0 warnings 4 CPU seconds
```

Now type “`dir`”, and you will see that you have more files in your directory:

```
1 drwx----- 2 pjg25 csda 512 Sep 14 15:24 ./
1 drwx----- 3 pjg25 csda 512 Sep 14 15:24 ../
792 -rw----- 1 pjg25 csda 796250 Sep 14 15:21 exer1.spss
19 -rw----- 1 pjg25 csda 19062 Sep 14 15:25 exer2.lis
9 -rw----- 1 pjg25 csda 7209 Sep 14 15:21 exer2.sas
9 -rw----- 1 pjg25 csda 8951 Sep 14 15:21 exer2.spss
184 -rw----- 1 pjg25 csda 180100 Sep 14 15:25 exer2.spssfile
1 -rw----- 1 pjg25 csda 104 Sep 14 15:21 exer3.spss
1 -rw----- 1 pjg25 csda 18 Sep 14 15:21 rmal
1 -rwx----- 1 pjg25 csda 37 Sep 14 15:21 sps*
```

So that you can edit the file `exer3.spss`, issue the command: `pico exer3.spss`

Move your cursor down to the line that looks like:

```
get file = '/tmp/YOURID/csda/exer2.spssfile'
```

This line asks SPSS to read the systems file created when we ran the program “`exer2.spss`”. Edit the file by substituting your own user id (so that you properly define the full path name of the file) for “`YOURID`”. Exit and save the file (`^X`) and `Y`. Now run the program by typing: `sps exer3 &`

When it is completed you will see the line:

```
End of job: 5 command lines 0 errors 0 warnings 0 CPU seconds
```

**Running a SAS job** - Please refer to the document “Using SAS on the UNIX” and other help documents on SAS on CSDA’s web page.

Take a look at the file “`exer2.sas`” by typing: `pico exer2.sas`

Page through the file by using `^v`. If you have used SAS on the IBM or on personal computers, this should look very familiar to you. Now exit from pico with `^X` and run the program by executing the command: `sas exer2 &`

After the program is finished, type “`dir`” and you will see that there are two additional files now -- `exer2.lst` and `exer2.log`. As on the IBM mainframe, SAS creates separate listing and log files.

### E-Mail - PINE

A software available for e-mail is “pine.” It is pretty simple to use, but can do a lot. Unlike some mail software, pine can send and receive attachments easily. Pine also has a nice address book system. The commands within pine are not case sensitive. The menus are quite clear, and I will not be explaining most of the options. But, pine, which is related

to the editor “pico”, shares some annoying features with “pico”. Specifically, using your Windows tools to cut text does not work. And, using your Windows tool to paste does not work well.

To get into pine, type “pine” in lower case. Pine is menu driven, and the following menu will come up.

```
PINE 3.93      MAIN MENU                               Folder: INBOX  53 Messages

?      HELP                - Get help using Pine
C      COMPOSE MESSAGE     - Compose and send/post a message
I      FOLDER INDEX        - View messages in current folder
L      FOLDER LIST         - Select a folder OR news group to view
A      ADDRESS BOOK        - Update address book
S      SETUP               - Configure or update Pine
Q      QUIT                - Exit the Pine program

                [Folder "INBOX" opened with 53 messages]
? Help          P PrevCmd          R RelNotes
O OTHER CMDS L [ListFldrs] N NextCmd  K KBLock
```

Once you get into pine, type “a” to get into your address book. You should see the following menu at the bottom of the screen. What else you will see depends on if you have any names in your address book.

```
? Help      < Main Menu      P PrevEntry      - PrevPage @ AddNew      C ComposeTo
O OTHER CMDS > [View/Update] N NextEntry  Spc NextPage D Delete    W WhereIs
```

Type “@” to add a new name. Put in your own name and e-mail address.

```
Nickname    :me
Fullname    :Lastname, Firstname
Fcc         : (put something here if you want mail for this entry to go into a special
folder)
Comment     :phone, any information you want
Addresses   :YOURID@csc.albany.edu
```

Fill in the fields just like you would in the composer.  
To form a list, just enter multiple comma-separated addresses.  
To add to a list, use the View/Edit cmd instead of the AddNew cmd.  
It is ok to leave fields blank. Press ^X to save the new entry.

```
^G Get Help  ^X eXit/Save ^R RichView  ^Y PrvPg/Top ^K Cut Line
^C Cancel    ^D Del Char  ^V NxtPg/End ^U UnDel Line ^T To AddrBk
```

After you have finished the entry, press “^X” (which, remember, is holding down the Ctrl key and then touching the X or x key). Pine will ask you if you want to save the entry, type Y or y. Pine stores addresses in a file in your home directory called .addressbook.

You can also create a mailing list. You would get into your address book, and type “@” to add a new address.

```
Nickname    :list1
...
Addresses   :user1,user2@cnsvox.albany.edu,nick1
```

You can enter either nicknames that are already in your address book, USERIDs that are on the SUNY UNIX, or full addresses for people not on the SUNY UNIX. The list of names should be separated with commas.

To send mail to people or mailing lists in your address book, you can either go into the “Compose Message” and enter the nickname in the “To:” field, or, go into your address book, highlight a name or list, and then type “C” for “compose to”. Send a message to “me” (your own user ID). After you have sent it, go to your folder list, and look at your “INBOX”. Within a couple of minutes, you should have a message from yourself. Push the return key when the message is highlighted, and you will be able to read the message you sent to yourself. Look at the bottom of the screen, and you will see the following menu.

```
? Help      < MsgIndex  P PrevMsg      - PrevPage D Delete      R Reply
O OTHER CMDS > ViewAttch N NextMsg      Spc NextPage U Undelete  F Forward
```

Type “O” or “o”, for “other cmds” and you will see the following menu.

```
? Help      M Main Menu  L ListFldrs  C Compose    % Print      S Save
O OTHER CMDS Q Quit Pine  G GotoFldr   W WhereIs    T TakeAddr   E Export
```

Note that you can type “S” for save. By default, it will save it in a subdirectory in your home directory called Mail, in a file called “saved-messages”. You can choose to save it in a different file if you want, by giving another name at that point. In this case, save the message in a file called “practice”. When you save a message, it is marked for deletion. When you exit from pine, pine will ask you if you want the messages marked for deletion deleted. You can remove the deletion mark by going to the index of you inbox and typing “u” by the message. You can delete a message without saving it by typing “d”.

Now practice composing a note, and including an attachment. You can attach binary files, or ASCII files -- pine will detect the type and deal with them appropriately. It is possible to FTP binary word processing files to the UNIX, and send them as an attachment using pine -- if the person you are sending mail to can also deal well with attachments (not all mail programs can). It is important for you and the person you send these files to test the process before you count on it. For this exercise, though, we will just use an ASCII file.

Type “C” to compose a letter. In the address field, put your own userid. Now go down to the “Attachments:” line, and type “^J”. Now type “^T”. You will see the files in your directory. Highlight the directory “Mail” and hit return. Highlight the file called “practice” that you created in the previous step, and hit return. It is now an attachment to the letter. Write a note to yourself if you want, and then send the mail by typing “^X”. Practice viewing the attachment after you receive the mail you just sent to yourself. Use the menu to figure out the proper commands.

~~~~~  
PLEASE NOTE: The remaining sections are for reference - they are not “hands-on” exercises.

FILE AND DIRECTORY PERMISSIONS AND OWNERSHIP (For Sharing Files)

This section will give a brief overview of how to share files with others. If you want more detailed information, see [The UNIX Handbook](#) (which can be purchased at the bookstore, or is available on CSDA’s web page referred to above). File permissions on UNIX systems are flexible and tricky. By default, only the owner of a file can read or edit it. But, it is possible to allow others to read and/or write to your files. Before a file can be read by someone, the directory with the file in it and all of the directories above that must be made readable to that person. For example, for someone to read a file in the directory /home2/generic/setclass/csda/, the directories /home2, home2/generic, and /home2/generic/setclass/ and /home2/generic/setclass/csda/ must all be readable by that person. Users can be members of more than one “group”. Assignment to groups is done by administrators of the UNIX cluster. If you need to share files with some users, but keep them private from other users, a group can be created for you. You would send a note to “helpdesk” to request the creation of a group.

The following set of commands will grant read and write permission to a directory for the group that the directory is associated with. To change the permission so that everyone (all) could read and write to the directory, “g+” would be changed to “a+” in the last three commands.

```
chmod u-l  dirname  unlock the directory (required)
chmod u+t  dirname  put the sticky bit on, so only you can delete files you don't own
chmod g+x  dirname  give execute permission to your group (required)
chmod g+r  dirname  give read permission to your group
chmod g+w  dirname  give write permission to your group
```

To change the group of a directory or file, issue the following command (substituting “newgroup” with the name of the group you want for the file, and “filename” with the file or directory that you want changed).

chgrp newgroup filename

After the directory is made readable or writeable by others, read or write access can be given to files within the directory, or they can be left private.

If you type `dir` you can see who owns which files and what the permissions of them are. For example:

	Permissions	Owner	Group	Filesize	Date modified	filename
1	-rw-r-----	1 pjpg25	csda	272	Jan 29 1996	adjust.sas
2392	-rw-r-----	1 pjpg25	csda	2441216	Feb 4 09:13	amod6.ssd01
2	-rw-r-----	1 pjpg25	csda	1525	Jan 29 10:46	catmod4.sas
50	-rw-----	1 pjpg25	csda	51200	Mar 30 12:34	chap2.doc

The column on the left displays the permissions for the files listed here. The first symbol is a hyphen (-) if the filename represents a file or the letter d if the filename represents a directory. The first group of three symbols (rwx --- ---) stands for the permissions granted to the owner of the file. The second group of symbols (--- rwx ---) stands for the permissions granted to the group that the owner belongs to (type `id` to see what group the file is associated with). The last set of symbols (--- --- rwx) describes the permissions granted to everybody else on the computer.

Each symbol is either a hyphen or the appropriate letter for that space (an r, w, or x).

An r (r--) stands for read permission. If there is an r instead of a hyphen, then the file can be read. If there is a hyphen, then the file cannot be read. A w (-w-) stands for write permission. An x (--x) stands for execute permission. Programs or shell scripts need to have this permission set or the computer will refuse to run them. Directories need to have execute permission for you to `cd` into them.

Some examples of file permission settings:

-rw-rw-rw-	Everyone can read and write this file.
-rw-r--r--	Everyone can read this file, but only the owner can change it.
-rwx-----	Only the owner can read, write, and execute this file.
-rwxr-xr-x	This combination is usually seen for a program that everyone needs to be able to read and execute but only the owner (which is probably the system administrator) can change or delete.

Granting or Removing Permission - Some examples

<code>chmod u+x a.out</code>	This makes the program <code>a.out</code> executable for the user only.
<code>chmod ugo+r news</code>	This lets anyone read the file called <code>news</code> . This is the same as typing <code>chmod a+r news</code> .
<code>chmod g-r *</code>	Removes read permission to group for all files in directory.

SUGGESTIONS FOR SPACE MANAGEMENT (IF YOU ENCOUNTER SPACE PROBLEMS)

UNIX Temporary Directories and “Saving Space” - Students only have 3 megabytes of disk space in their home directory, and some of this space is required for configuration files which you cannot delete. You may find that you do not have enough space in your home directory to store all of the files you need to store. In this section, we will discuss ways of managing that problem. You can work in temporary space (`/tmp` on `eve`), and store files in a compressed format on your home directory or on a PC when you are not using them. For each of the methods discussed, if you try to move something to your home directory that takes up more space than you are allowed, you may receive an error message. We recommend that you compress your files before moving them to your home directory. At the end of your session, enter the command “`quota -v`” to make sure that you have not exceeded your disk space limit on your home directory. The first step for the methods we will discuss is to compress your files. Start with the command:

```
gzip -r /tmp/YOURID/* &
```

(`-r` is a common modifier to UNIX commands that means “recursive” – through subdirectories).

A. Moving a directory, subdirectories, and contents to your home directory -- In our example, a user has created a directory in /tmp named YOURID. After the files have been gzipped, the directory, files and subdirectories can be moved to your home directory with the following command: `“mv /tmp/YOURID ~/”`. (“~/” is shorthand in UNIX for the home directory of the account you are logged onto. It is an easy way to refer to your home directory.) When you are ready to use the files again, from your home directory, you could issue the command `“mv ~/YOURID /tmp/”`. Restore the files with `“cd /tmp”`, then `“gunzip -r YOURID ”`. You could save additional space by deleting files that you did not need, or that could be easily recreated (such as output files from SAS and SPSS programs).

B. Use “tar” to create an archive, compress it, then move it, or FTP it to a PC - Assume you have created a directory in /tmp called YOURID. This directory also contains sub-directories and files. All of the files and directories in the directory “YOURID” can then be put into a single file using the “tar” command. From the directory /tmp, you would issue the command:

```
tar cvf YOURID.tar YOURID
```

A file called YOURID.tar will be created that will hold all of the files and directories in directory YOURID. We recommend that you use the .tar extension to identify tarfile archives. You can then compress the tarfile using the gzip command, and copy (cp) or move (mv) it to your home directory with the following commands.

```
gzip YOURID.tar &  
mv YOURID.tar.gz ~/
```

When you are ready to use the files again restore all of the files onto /tmp with the following commands:

```
cd  
mv YOURID.tar.gz /tmp/  
cd /tmp  
gunzip YOURID.tar.gz  
tar xvf YOURID.tar  
gunzip -r YOURID.
```

If your home directory does not have enough room to store the compressed tarfile, you can FTP the file (as a binary file) to your PC, and store it there. Be sure to test this to make sure that you have all of the parameters correct when you FTP before relying on it!

A Few Useful Commands - Following is a list of commonly used commands, their meaning, and an example of syntax. Remember that the UNIX is case sensitive. For some commands, an asterisk can be used as a wild card on the UNIX.

Command	Meaning	Examples of Syntax
man	show man ual pages for the program named (if available)	man chmod
&	run in background This symbol, following most commands, will cause the job to run in background	sas progname & spss progname &
cd	change directory	cd /malthus2/csda/help
pwd	tell me present working directory	pwd
mkdir	make a new directory	mkdir newdir
rmdir	remove a directory	rmdir newdir
cp	copy a file copy to different directory	cp old new cp old /full/path/new
rm	dangerous! remove file	rm oldfile
ps -aux	give information current jobs	ps -aux
kill -9 jobnum	kill a job that is running You can only kill jobs that you own.	kill -9 3718
dir	give information about files in the directory tell about files ending in .lst	dir dir *.lst
ls	give concise directory list	ls
groups	find out groups a user is in	groups userid
Ctrl c	Stop a process	Press Ctrl key, and tap C key
!!	repeat my last command	!!
h	tell me history of commands	h
!# (a number)	repeat a command shown in history (h from above)	!3
gzip	compress a file with the gzip utility. The extension will be .gz	gzip filename
gunzip	uncompress a file "gzipped" file	gunzip filename.gz
compress	compress a file with the compress utility. The extension will be .Z	compress filename
uncompress	uncompress a compressed file	uncompress filename.Z
zmore	show me a compressed file space key for next page; q for stop	zmore filename.gz
more	show me a file space key for next page; q for stop	more filename
script	create a file of my conversation with the computer. Close the file with the command exit	script filename exit
reskel	update/recreate .cshrc, .login, other configuration files.	reskel
lp	print a file	lp -c -dlp filename
cal	show a calendar for 7/99	cal 7 99
who	show who is logged on	who
w	show current processes	w
	pipe - used with other commands	who more
grep	search	ps -aux grep yourid
find	find a file beginning with xxx	find . -name "xxx*" -print
*	wild card - can be used with a variety of commands	chmod g+r *
~/	shorthand for home directory	cp file1 ~/file1

C:\all\HELP\CLASSA.wpd July 3, 2000