THE Unix COOK BOOK

Campus Computing, University of Missouri-Columbia

This document summarizes the most generally useful commands of the Unix computer systems available at Mizzou. For more detailed information consult the Unix man pages available on all the Unix operating systems. Specific Unix systems often have considerable on-line documentation besides the Unix man pages. Consult the system-specific documentation. Also, on SHOWME, considerable information is in gopher.

Examples in this Cook Book will show commands on **showme.missouri.edu**. Called SHOWME for short.

The description of the commands follows the convention of showing them in the `Courier` font. <u>For Unix commands and file names: **capitalization is significant!**</u> Commands and file names will be shown in their appropriate case.

The Unix Operating System is Case Sensitive!!

The following Unix systems are available at public computer sites, by dial-up or via telnet or ftp.

Unix System	MU host names	Sites Available
AIX (IBM RS/6000)	showme.missouri.edu	Most sites; via telnet
Silicon Graphics	sgi*.missouri.edu	127 Physics 124 GCB
NeXT	muebnx*.missouri.edu mugcnx*.missouri.edu muphnx*.missouri.edu	17 Engineering Complex West 222 GCB 127 Physics

* Each workstation has a number associated with it and is part of the host name.

University Rules Regarding Computing

Access is granted only to students currently enrolled in any MU course or to faculty members.

The MU rules regarding proper use of computing say, in general, conserve resources and do not use any resource without proper authorization. No one may use a student user ID except the student with the corresponding student number. Those found abusing computing resources are denied access to them and may face suspension or other censure.

More details regarding MU's User ID Policy, Rules of Computing and the Ethics of Computing are available in `gopher` in the `3. Guidelines/` section..

Logging On

1. Telnet to the machine of your choice. If you need assistance, ask one of the User Consultants at one of the principle computing sites.

2. The SHOWME login prompt will look like:

```
showme.missouri.edu
login:  _
```

Your login user ID will be a lower case 'c' followed by your student number if you are a student. If you have a faculty staff user ID consult the documentation mailed to you by the Campus Computing Help Desk for your login user ID and default password.

You will then be prompted to enter your logon password.

```
showme.missouri.edu
login:  c123456
c123456's Password:  _
```

If you have a student user ID then your default password will be your birthdate in the form of *yymmdd.* For example: 740607 if your birthdate is June 7, 1974. The first time you login the system will prompt you for a new password. You will prompted to type in your new password twice (to ensure against typos) so both the computer and you think you have the same password. Your password will not be displayed when you type it in.

```
showme.missouri.edu
login:  c123456
c123456's password:
You are required to change your password.
Please choose a new one.
c123456's New Password:  _
```

Choose a password that is easy for you to remember but one that isn't easy for someone to guess.

- *DO* use a password with numbers or punctuation.
- *DO* use a password that is easy to remember. Never write a password down anywhere!
- *DO* use a password that you can type quickly, without having to look at the keyboard.
- **DON'T** use your login name in any form.
- **DON'T** use your first or last name in any form.
- **DON'T** use your spouse's, child's or pet's name.
- **DON'T** use other information easily obtained about use. This includes license plate numbers, telephone numbers, social security numbers, the brand of your automobile, the name of the street you live on, etc.
- **DON'T** use a password of all digits, or all the same letter.
- **DON'T** use a word contained in an English or foreign language dictionary.
- **DON'T** use a password shorter than 6 characters.

Remember **Unix is case sensitive.** The password: `truman99` is not the same as: `Truman99` or even the same as: `TRUMAN99`. More information about password selection is available in `gopher`.

```
showme.missouri.edu
login:  c123456
c123456's password:
You are required to change your password.
Please choose a new one.
c123456's New Password:
Re-enter c123456's New Password:  _
```

Announcements may appear after you login. These login messages will tell you about the machine's availability, system maintenance, software changes, or any other information that you may need. Read these messages every time you login!

```
showme.missouri.edu
login:  c123456
c123456's password:
You are required to change your password.
Please choose a new one.
c123456's New Password:

********************************************************
* showme.missouri.edu        aix 3.2.5 on rs/6000 590 *
* -- watch this space for announcements        012794 *
* -- utilize "gopher" to access usage info.    012794 *
********************************************************
showme:c123456> _
```

This is a shell prompt -- here is where you may enter your commands.

Logging Off

To terminate your Unix session press `^D` (Control-D) until your session ends. This won't work if you have previously set `ignoreeof`. If this case, you must type `logout` or `exit`.

Assistance

On SHOWME extensive help is available via *gopher*. Type in `gopher` from the shell prompt and you'll get a menu that looks something like:

```
                    Internet Gopher Information Client v2.0.16

                      Home gopher server:  gopher.missouri.edu

-->_ 1.  About the SHOWME gopher /
     2.  What is SHOWME? /
     3.  Guidlines /
     4.  How To ... /
     5.  News & Weather /
     6.  The Internet /
     7.  Around Columbia, MO. /
     8.  Feedback & More Information /
     9.  Campus Information & Gopher Servers /
     10. Search the SHOWME gopher menus using jughead <?>



Press ? for Help, q to quit
```

To ask for a list of Unix commands involving some keyword, enter a command like this:

        man -k *keyword*

or

        apropos *keyword*

To get a detailed description of a specific Unix command or facility, enter something like this:

        man *command*

Specific Unix systems often have considerable on-line documentation besides the Unix man pages. Consult the system-specific documentation.

Control Characters

These are the default control characters. You may change control characters with the `stty` command. `^` means hold down the *Control* key while pressing the specified key.

`^C`	kill the current process or line being entered
`^D`	end of line. If you are reading a file from the terminal, this will end it. If you are at a shell prompt, it will log you out unless you have set `ignoreeof`.
`^Z`	stop current job, but leave it around. For more information see `man csh`

Command Format

Unix commands are typically of the format:

```
command [-options] arguments [redirection] [&]
```

Where `command` is the file name of a program. The `PATH` variable defined by the `set` command specifies directories that will be searched for this name. An explicit path name can be specified for a command, for example: telnet can be invoked by: `/usr/ucb/telnet`

`options` are typically single letters with a "`-`" before them. Options may be specified individually, for example:

```
ls -l -g -R
```

Or usually they can be grouped:

```
ls -lgR
```

`arguments` are normally file names, separated by spaces, but can be other things. File names can include wildcards `*` and `?`. Special characters can be quoted with the backslash `\` or single quotes `' '`.

The terminal is the default standard input, standard output and standard error message output for most commands. To send output to or take input from another device, use `redirection`:

`<`	*inputfile*	read standard input from file
`>`	*outputfile*	write standard output to file
`>>`	*outputfile*	append standard output to file
`>&`	*errorfile*	write error message to file
`|`	*command2*	pipe output as input for command2

`&` at the end of a command line causes it to be done in the background. You can type other commands while it is being processed. See below for session control and job management commands.

Session Control and Job Management

Brackets around a word [ ] indicate that this is an option of the command. A word in italics indicates you need to supply what is in italics.

|	
`clear`	Clear terminal screen.
`jobs`	List stopped and background jobs. Only valid in C Shell.
`bg [%job]`	Continue job in background. [Default: current job]. Only valid in C Shell.
`fg [%job]`	Continue job in foreground. [Default: current job]. Only valid in C Shell.
`stop [%job]`	Stop job running in background. [Default: current job]. Only valid in C Shell.
`%job [&]`	Continue job in foreground [in background if `&`]. Only valid in C Shell
`nice command &`	Run job in background at low priority. See also `at` command.
`time command`	Show how much CPU and real time *command* uses
`history [n]`	List most recent *n* commands. If you want to do this, you must previously `set history = `*m*, where *m* is the number of commands you want remembered. Only valid in a C Shell.
`ps [options]`	Show a process. The default is show only yours.
`-a`	All processes controlled by terminal.
`-g`	Show group leaders (top level processes).
`-tx`	Processes on `tty x`, *e.g.* `-tp2` or `-tttyp2`
`-u`	User oriented output. Processes for a specific user.
`-x`	Show even processes with no terminal.
`kill 123`	Kill process 123 [system-wide numbering, use "`ps`"].
`kill %1`	Kill job 1 [your process 1, use "`jobs`"].
`at time [commandfile]`	Run program/script at a later time. **e.g.:** `0300    0300  friday 0300 jan 25`

File and Directory Names
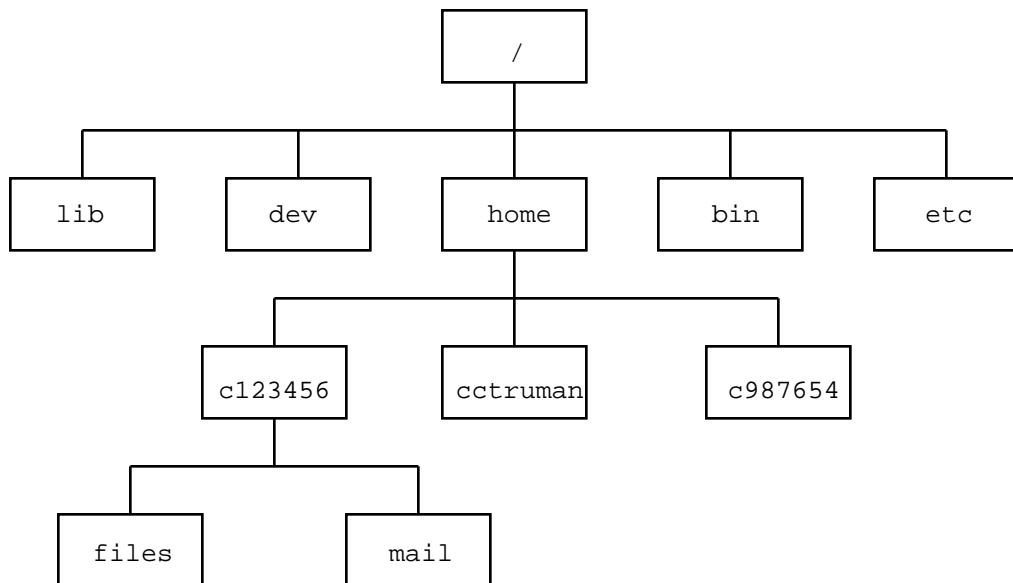
Each file name may contain any of the ascii characters and be up to 255 characters in length.

Capitalization counts! File `stuff` is different than file `Stuff`.

If a filename contains characters special to a shell (such as: `>` `|` `&` `!` `?` or *blank*) you must enclose it in double or single quotes. For example:

        `'My  #1 File'` or `"/tmp/huh?"` or `'Read\ Me\!'`

Unix files are arranged in a hierarchical structure, much like DOS files are stored. Path names are the vehicles for moving up and down the directory tree. The directory tree starts with `/` (called `root`).

```
                              ┌─────────┐
                              │    /    │
                              └─────────┘
          ┌───────────┬───────────┼───────────┬───────────┐
      ┌───────┐   ┌───────┐   ┌───────┐   ┌───────┐   ┌───────┐
      │  lib  │   │  dev  │   │ home  │   │  bin  │   │  etc  │
      └───────┘   └───────┘   └───────┘   └───────┘   └───────┘
                          ┌───────────┼───────────┐
                      ┌─────────┐ ┌──────────┐ ┌─────────┐
                      │ c123456 │ │ cctruman │ │ c987654 │
                      └─────────┘ └──────────┘ └─────────┘
                    ┌───────────┴───────────┐
                ┌───────┐               ┌───────┐
                │ files │               │ mail  │
                └───────┘               └───────┘
```

If I wanted to reference absolutely a file in c123456's files subdirectory it would look something like:

        `/home/c123456/files/assgn1.c`

The total path name length must not exceed 1024 characters.

Your home directory will be: `/home/userid`. If you are user ID `c123456` then your home directory will be: `/home/c123456`.

`.`	The current directory.
`..`	The directory above the current one.
`~c987654`	Home directory for user ID `c987654`
`~`	Your own home directory.

Directory and File Management

`cat files`	conCATenate: Copy `files` to standard. output as one file.
`cd directory`	Change Working Directory.
`chgrp group files`	CHange GRouP id for files.
`chmod who±level files`	CHange MODe for files; set security access.
	`who` can be:
u	User
g	Group
a	All
	± can be:
+	Add this permission level.
−	Remove this permission level.
	`level` can be:
x	executable for files, searchable for directories
X	extend existing execute permissions
r	read permission
w	write permission
`cp infile outfile`	CoPy file. See also `mv` and `tar`.
`cp [options] files directory`	CoPy several files into a different directory.
-p	Preserve original permissions, dates.
-i	Interactively confirm if will overwrite an old file
-r	Recursively copy all files in directories.
`compress [filename]`	Compress a file to save space.
`uncompress [filename.Z]`	Uncompress a file.
`diff file1 file2`	show DIFFerences between files or directories.
-b	ignore Blank spaces.
-c	show more of the content.
`df`	shows total amount of Disk space Free.
`du [files]`	show Disk Usage [Default: all directories below the current directory]
-s	Show sum for each file.
`find [start-directories] [characteristics]`	search for files with various characteristics.
-name '*stuff*'	find files with `stuff` in name.
-mtime +5	find files modified more than 5 days ago.
-mtime -5	find files modified less than 5 days ago.
-size +1024c	find files bigger than 1K bytes.

`-print`	find full file names meeting previous criteria.
`-ls`	find file names and other information.
`-exec rm {} \;`	Remove the files meeting previous criteria.
`flip [`*`files`*`]`	convert carriage-return/line-feed and other formats.
`-u`	convert to Unix format from DOS format.
`-m`	convert to MS-DOS format from Unix format.
`head [`*`files`*`]`	show first few lines of files. [Default: first 10 lines]
`ln `*`oldfile newfile`*	LiNk files: create an altername for a file.
`ln -s `*`fielspec name`*	Symbolic LiNk: *name* will refer to `filespec`.
`ls [files]`	LiST files [Default: all files in directory.]
`-a`	All files [Normally files beginning with . are not shown.]
`-l`	Long form.
`-t`	Time sorted - most recent first.
`-R`	Recursively look into directories.
`mkdir [`*`files`*`]`	Make new DIRectories.
`more [`*`files`*`]`	display files, stop when screen is full. [Default: std in.]
`mv oldfile newname`	rename or MoVe file(s).
`-i`	Interactively confirm if new name exists.
`pwd`	Print Working Directory.
`rm `*`files`*	ReMove files.
`-i`	Interactively confirms each one.
`-r`	Recursively remove contents of directories.
`rmdir `*`file`*	Remove an empty DIRectory
`tail [`*`file`*`]`	show last few lines of a file. [Default: last 10 lines.]
`tar [`*`op`*`] [`*`file`*`]`	Tape Archive and Restore. Used to unload an entire directory structure into a simple file for storage or sending.

DON'T FORGET: Unix Commands are Case Sensative.

File Information Provided by Command: `ls -lgi`

```
15794 d rwx --- --- 2 student  512 Apr  1 16:03 mail/
19103 - rw- --- --- 1 student  512 Apr  1 14:47 sasuser/
  601 - rw- --- --- 1 student 1024 Mar 21 11:05 class.ssd01
   84 - rw- rw- --- 1 student   62 Mar 15 07:57 howework.sas
  555 - rwx --- --- 1 student   36 Mar 14 13:32 forecast*
```

Type User Group Other

I-Node   Permissions   Links Group   Size   Date/Time   File Name

I-Node	Disk Location of file.
Permissions	Type of file and who can access it.
Type	d for directory
	l for symbolic link
	- for plain file
Access	r for read access
	w for write access
	x for execute file or search directory
Links	Number of hard links (synonyms) for file.
Group	The group associated with the file.
Size	Number of bytes. Direstories take multiples of 1024.
Date	Switches time to year for files older than six months.
File Name	Appended to name can be:
	/ for a directory
	* for an executable file
	@ for a symbolic link

Communicating with Others

Your electronic mail address is: *userID@showme.missouri.edu*

      For example: *c123456@showme.missouri.edu*
                           cctruman@showme.missouri.edu

There are two different e-mail systems available on SHOWME. They are Pine and Elm. Both systems are menu driven. You may choose whichever one meets your e-mail needs.

Pine

Pine is a menu based electronic mail program. To run the program, type `pine` from your Unix prompt. The first time you invoke pine, you will see the following screen.

```
  PINE 3.87   MAIN MENU                         Folder: (CLOSED)  0 Messages


                        Welcome to Pine...

   a Program for Internet News and Email.  Pine offers the ability to:
     -Access local and remote message folders using a simple user-interface
     -Send documents, graphics, etc (via the MIME standard for attachments)

   COMMANDS IN PINE:  The last two lines on the screen tell you what
   commands are available for the current situation.  Usually there
   are more commands than can be shown on two lines, so use the "O"
   key to see what OTHER COMMANDS are available.  The "O" is optional;
   it is not necessary for a command to be visible before using it
   (except when Pine is used in function-key mode; then F2 is required.)


   PINE CONFIGURATION:  Pine will create a default configuration file,
   .pinerc, in your home directory.  You may edit this file to select
   various options.  It will also create a "mail" subdirectory for your
   saved-message folders.  PLEASE type "?" from the Main Menu for help.
           PINE is a trademark of the University of Washington.
Type any character to continue :
```

From this point on, you'll see the pine main menu screen:

```
   PINE 3.87   MAIN MENU                          Folder: INBOX  8 Messages


       ?     HELP                -  Get help using Pine

       C     COMPOSE MESSAGE     -  Compose and send a message

       I     FOLDER INDEX        -  View messages in current folder

       L     FOLDER LIST         -  Select a folder to view

       A     ADDRESS BOOK        -  Update address book

       S     SETUP               -  Configure or update Pine

       Q     QUIT                -  Exit the Pine program




    Copyright 1989-1993.  PINE is a trademark of the University of Washington.
                  [Folder "INBOX" opened with 8 messages]
  ? Help                      P PrevCmd                   R RelNotes
  O OTHER CMDS L [ListFldrs] N NextCmd                    K KBLock
```

This screen format is common through-out the pine program. Up at the top of the screen, pine will display the version number of the program, followed by the menu option you are currently working in. At the top right of the screen is the current folder and how many messages reside there. In the center of the screen are the options from this menu, and at the bottom the single letter commands for pine are listed.

To Send Mail

- Press **C** from the main menu of pine to compose a message.
- Enter names into **To:** field.
  (You can use nicknames that you have set up in the Address Book)
- Press **<tab>** to move on, fill in other fields as desired (especially the **subject** field).
- Press **<tab>** to move to the body of the note.
- enter the text of the note.
- Press **^X** (control-X) to send, then **Y** to confirm.

Editing Your Note

- Words will autowrap as you type.
- To insert letters, words, or sentences, move to where you want to insert & type.
- Backspace key will delete letters.
- Space will be added to the note as needed (as you type).
- Use **^J** (control-J) to rejustify text after editing.
- Spell check by pressing **^T** (control-T).

To Look at Incoming Mail

- Press **L** from the main menu of pine to list available folders.
- Press **<tab>** to move to '**inbox**', press **<return>** to select.
- Use **<tab>** to move to the message you wish to read, press **V** to view the note.
- When finished with that note, press **<tab>** or **N** to view the next note
- When finished reading mail, press **M** to return to the Main Menu

Replying to Incoming Mail

- While viewing the note press **R** (for reply).
- You may include the original text of the note by responding **Y** to the prompt as you begin your reply.
- Type in your response to the note.
- When finished with that note, press **^X** (control-X) to send the note.

Managing your Mail

- You may press **I** from the Main Menu to view the notes in your current folder.
- You may press **L** from the Main Menu to list and change to your other mail folders.
- You may press **D** to delete a note.
- You may press **U** to un-delete a note.
- While reading notes, you can save them by pressing **S**.
  You will be prompted to save the message in the "**saved-messages**" folder.
  The message will then be marked for deleting from your inbox.

Creating/Editing the Address Book

- Press **A** from the Main Menu to go to the Address Book option in Pine.
- Press **A** to add a name to the address book.
  Enter their full name as prompted (Last, First)
  Enter a nickname to use in Pine
  Enter their email address
- To edit a field, move there with the arrow keys or tab and press **E**.
- You may press **D** to delete a nickname.

More Details

- More details on Pine are included on-line in gopher.
- ---> 4. How to ... /
-       ---> 4. Electronic Mail: the mail programs /
-             ---> 2. Pine: help on Pine /

Elm

Elm is a menu based electronic mail program. To run the program, type `elm` from your Unix prompt.. The first time you invoke elm, you will see the following message.

```
Notice:
This version of ELM requires the use of a .elm directory in your home
directory to store your elmrc and alias files. Shall I create the
directory .elm for you and set it up (y/n/q)? n
```

Enter **Y** at the prompt to set up elm. It will be followed by:

```
Great! I'll do it now.

Notice:
ELM requires the use of a folders directory to store your mail folders in.
Shall I create the directory /showme/cctruman/Mail for you (y/n/q)? y
```

Enter **Y** at the prompt to continue to set up elm. When it's finished, it will be show you the main Elm screen:

```
        Mailbox is '/usr/spool/mail/cctruman' with 0 messages [ELM 2.4 PL23]












   You can use any of the following commands by pressing the first character;
d)elete or u)ndelete mail,  m)ail a message,  r)eply or f)orward mail,  q)uit
     To read a message, press <return>.  j = move down, k = move up, ? = help

Command:
```

To Send Mail

- Press **m** from the main screen.
- Enter the Email address when it asks you '**Send the message to:**'.
  (You can use nicknames that you have set up with aliases)
- Enter the subject when it asks you '**Subject of message:**'.
- When it asks you '**Copies to:**', you can include the Email addresses or nicknames of people you wish to receive complimentary copies of this mail. The default value is no copies sent.
- Elm will invoke it's default editor (usually vi). Enter your message, ending with **:wq**
- Press **s** to send.

To Look at Incoming Mail

- Use the arrow keys to move about messages list.
- Press return to view a message.

Replying to Incoming Mail

- While viewing the note press **r** (for reply).
- You may include the original text of the note by responding **y** to the prompt as you begin your reply.
- Write your reply, ending with **:wq**. Press **s** to send.

More Details
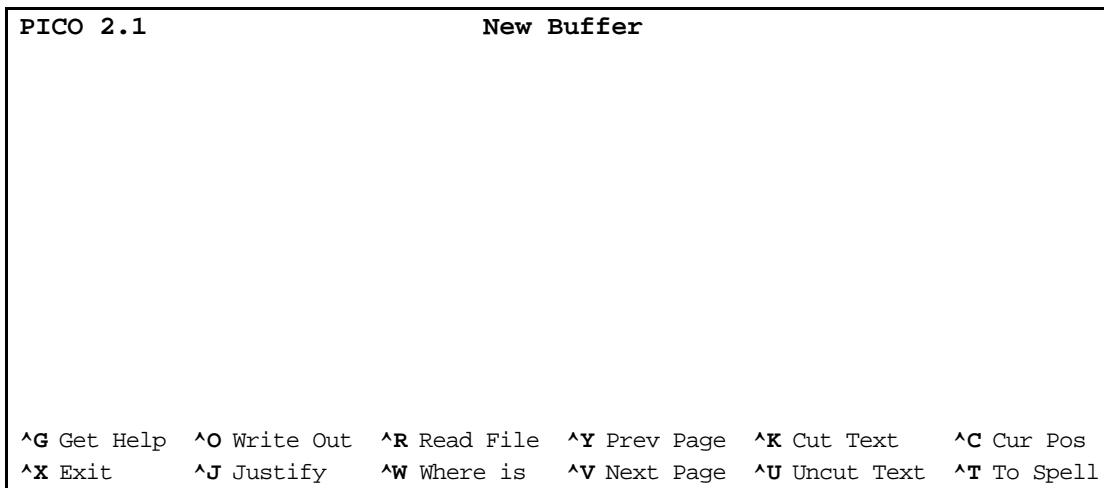
- More details on elm are included on-line in `gopher`.
- `---> 4.  How to ... /`
- `      ---> 4.  Electronic Mail:  the mail programs /`
- `            ---> 1.  Elm:  help on Elm /`

Creating and Modifying UNIX Disk Files

Three of the most popular editors on SHOWME are; `pico`, `vi` and `x`. `Pico` is the easiest for beginners, `x` is almost identical to XEDIT on MIZZOU1 CMS and `vi` is a Unix standard.

The Pico Editor Reference

Pico is a simple, display oriented text editor. Commands are displayed at the bottom of the screen and context-sensitive help is provided. by pressing **^g** (Control-G).   As characters are typed they are immediately inserted into the text. Editing commands are entered using control-key combinations.   To start editing a new file enter **pico** from the SHOWME prompt. The screen should look something like:

```
PICO 2.1                        New Buffer








^G Get Help   ^O Write Out   ^R Read File   ^Y Prev Page   ^K Cut Text      ^C Cur Pos
^X Exit       ^J Justify      ^W Where is    ^V Next Page   ^U Uncut Text   ^T To Spell
```

Now you are ready to begin typing in your file. You do this by simply typing in your text. When you finish typing your file press **^o** (Write Out). Pico will prompt you to name yur file. Filenames should be fairly descriptive and meaningful to you and to SHOWME. For example, if you are entering in a SAS program you might want to name it something like `homework1.sas`. See **File and Directory Names** earlier in this Cookbook for more information on naming files.

If you want to edit an existing file, rather than creating a new one, enter **pico** ***<filename>*** from the SHOWME prompt. Where *<filename>* is the name of the file you want to edit.

Control Key Commands

Ctrl-f	Moves cursor forward a character
Ctrl-b	Moves cursor backward a character
Ctrl-p	Moves cursor to the previous line
Ctrl-n	Moves cursor to the next (following) line
Ctrl-a	Moves cursor to the beginning of the current line
Ctrl-e	Moves cursor tot he end of the current line
Ctrl-v	Moves cursor forward one page
Ctrl-y	Move cursor backward one page
Ctrl-w	Searches for a string of text
Ctrl-l	Refreshes the screen display
Ctrl-d	Deletes the character at the cursor position
Ctrl-k	Cuts selected text
Ctrl-u	Pastes the text block, which was last cut, at the current cursor position.
Ctrl-r	Inserts an external file at the cursor position
Ctrl-o	Saves the current file
Ctrl-x	Exits Pico
Ctrl-i	Inserts a tab at the current cursor position
Ctrl-j	Justifies the current paragraph
Ctrl-t	Invokes the spelling checker
Ctrl-g	Help
Ctrl-t	Justify text
DELETE key	Delete character to the left

vi Editor Reference

vi is a basic, full screen editor common to most Unix operating systems. The program is actually a combination of vi (the full screen) and ed (the background macros). You invoke the 'ed' section of the vi editor when you use the ":" commands. A quick reference is provided here.

REMEMBER: All Unix Commands are Case Sensitive!!

Abbreviations

`#`	a numeric value before the command. i.e. #G would mean 32G
`^X`	control-X
`<sp>`	space
`<cr>`	carriage return (the return or enter key)
`<string>`	a string of characters

Movement Commands

a number before a movement command repeats that command the number of times. If no number is typed before the command, the number is assumed to be 1.

`#h` (or left arrow)	left
`#j` (or down arrow)	down
`#l` (or right arrow)	right
`#k` (or up arrow)	up
`#$`	end of line
`#^`	first char of line
`#w`	move a word forward
`#b`	move a word backwards
`#G`	goto to line #
`#)`	sentence forward
`#(`	sentence backward
`#}`	paragraph forward
`#{`	paragraph backward
`^F`	screen forward
`^B`	screen backward

Searching

`/<string>`	search for next occurrence of <string>
`?<string>`	search for previous occurrence of <string>

Undoing Changes

`u`	undo latest change
`U`	Undo all changes to current line
`:q!`	quit vi without saving changes
`:e!`	re-edit the file

Deleting and Moving

Everything deleted is stored into a buffer. This is done by putting a " and a letter <a-z> before the delete command. The deleted text will be written into the buffer designated by the letter. To append text to a buffer, a capital letter <A-Z> is used.

`#x`	delete # characters after the cursor
`#X`	delete # characters before the cursor
`#dd`	delete # lines
`#dw`	delete # words
`D`	delete the rest of the line
`p`	put pack most recently deleted text before the cursor
`P`	put back most recently deleted text after the cursor

Inserting & Changing Text

All of these commands are finished by pressing the escape key `<esc>`.

`a`	append text
`i`	insert text
`#r`	replace # characters
`#R`	overwrite rest of line
`#J`	join # lines together
`:s/<string1>/<string2>`	substitute all <string1> with <string2>

Writing, Editing, & Quitting vi

`:q`	quit vi
`:q!`	quit vi without changing file
`:w`	write the file
`:w <string>`	write to file <string>
`:w >> <string>`	append to the end of file <string>
`:w! <string>`	overwrite the file <string>
`:wq`	write file and quit
`:f <string>`	set current filename to <string>
`:cd <string>`	change directory to <string>
`:r <string>`	read in file <string>

Miscellaneous Operators

:set number	display line number
:set all	show options and values
:*firstline,lastlines/original/new*/g	substitute original text with new (g means global replacement)

An example of the substitute command: `:1,$s/University of Missouri/MU/g`
Which says: Starting at line 1, for all lines, change all the occurences of "University of Missouri" to be "MU".

x Editor Reference

`x` is The Hessling Editor. It is intended to be similar to VM/CMS's XEDIT. It is a good editor for those of you who are already familar with editing files on MIZZOU1 and aren't quite ready to change to `pico` or `vi`. Most of the commands are similar to XEDIT's. There are some differences, however., the function keys being one of these; a table is below.

Standard x Function Key Assignments

Settings may vary. Enter `show` from the command line to see current settings.

F1	SOS Edit
F2	Split/Join
F3	Quit --If nothing has been changed, terminate editing.
F4	nop
F5	Undefined
F6	?
F7	Backward 1
F8	Forward 1
F9	File
F10	Shift Right 55 columns
F11	Shift Left 55 columns
F12	Do Prefix Command

Movement Commands

down arrow	move cursor down one line, scrolling if necessary
enter	execute command on command line or next line
left arrow	move cursor left one column scrolling if necessary
right arrow	move cursor right one column scrolling if necessary
up arrow	move cursor up one line scrolling if necessary (previous command if on command line)

Searching

These commands are performed from the command line.

/ <string>	search for next occurrence of <string>
-/ <string>	search for previous occurrence of <string>

Command Line Commands

These commands are performed from the command line.

?	Retreive last command
=	Execute last command line command
BOTtom	Go to Bottom of file
File	Save changes to file and exit x
GET [fileid]	Insert into current file contents of specified file
Quit	Quit x without saving
QQuit	Cancel any changes and exit x
REDRAW	Redraw screen after disturbance
Save	Write changes to file to disk but stay in x

Prefix Commands

These commands are performed in the numerical prefix area.

a or i	add blank line
/	make line current line
c	copy line
m	move line
d	delete line
"	duplicate line
p	prior to line; target for c and m
f	following line; target for c and m
>	shift line right
<	shift line left
cc	copy line block
mm	move line block
dd	delete line block
" "	duplicate line block
>>	shift line block right
<<	shift line block left

Programming Environments

Language	Extension	To Run:
FORTRAN	`.f`	`xlf fn.f`
C	`.c`	`xlc fn.c`
Pascal	`.p`	`xlp fn.p`
SAS	`.sas`	`sas fn`
Mathematica	`.ma`	`math`

Printing

The `lpr` command is used to print files in Unix. The format of the command is:

```
lpr [-options] filename(s)
```

The most common option you would want to specify is the printer destination, `-P`. (*Notice:* this is a *Capital* `P`) There is no default printer destination on SHOWME. If you wanted to print the file `homework1.sas` to the printer in the in the Heinkel Computing Lab the comand would be:

```
lpr -P fhrp1 homework1.sas
```

Following is a table of the most commly used printers on campus:

Printer Name	Location
aspr1	10 Arts & Sciences
bppr1	7 Middlebush
eepr1	17 Engineering Complex West
fhpr1	23 Heinkel Building
physpr1	127 Physics

Submitting Batch Jobs to the Compute Cluster

The Compute Cluster is available to users who have numerically intensive batch work. The cluster machines are shared via IBM's LoadLeveler Software. Software available on the cluster machines include: FORTRAN (including IMSL), SAS 6.09, Gaussian 92 and Mathematica.

Interactive shell scripts are available for easy job submission.

Shell Script Name	For Software:
fsub	FORTRAN and C
ssub	SAS 6.09
msub	Mathematica
gsub	Gaussian 92

There are four job queues setup.

Job Queue Name	CPU Time	Priority	Software
short	10 minutes	highest	all
medium	2 hours		all
long	200 hours		FORTRAN and SAS
g92	300 hours	lowest	Gaussian 92

Users can submit as many jobs as they wish, but only three jobs owned by any one user can run at the same time. This prevents any one use from flooding the queues, but still allows for all three machines being kept busy when only one user is active. More information about the Compute Cluster is available in gopher.

REMEMBER: All Unix Commands are Case Sensitive!!

Your Comments and Feedback

We are interested in your suggestions and comments! We all benefit from constructive comments about new services. In this developmental stage of general access Unix machines at MU, the unexpected is sure to happen. Work with us at Campus Computing to make this new student resource a success! Please e-mail your comments and suggestions to: helpdesk@showme.missouri.edu.