



This quick reference lists commands, including a syntax diagram and brief description. [...] indicates an optional part of the command. For more detail, use:

**man** *command*

Use **man** **tcsh** for the command language.

## 1. Files

### 1.1. Filename Substitution

Wild Cards	? *
Character Class (c is any single character)	[c...]
Range	[c-c]
Home Directory	~
Home Directory of Another User	~user
List Files in Current Directory	ls [-l]
List Hidden Files	ls -[l]a

### 1.2. File Manipulation

Display File Contents	cat filename
Copy	cp source destination
Move (Rename)	mv oldname newname
Remove (Delete)	rm filename
Create or Modify file	pico filename

### 1.3. File Properties

Seeing Permissions	ls -l filename
Changing Permissions	chmod nnn filename

**chmod** *c=p...[,c=p...] filename*

*n*, a digit from 0 to 7, sets the access level for the user (owner), group, and others (public), respectively. *c* is one of: **u**—user; **g**—group, **o**—others, or **a**—all. *p* is one of: **r**—read access, **w**—write access, or **x**—execute access.

Setting Default Permissions	umask ugo
-----------------------------	-----------

*ugo* is a (3-digit) number. Each digit restricts the default permissions for the user, group, and others, respectively.

Changing Modification Time	touch filename
Making Links	ln [-s] oldname newname
Seeing File Types	ls -F

### 1.4. Displaying a File with less

Run <b>less</b>	<b>less</b> filename
Next line	RETURN
Next Page	SPACE
Previous line	<b>k</b>
Previous Page	<b>b</b>

### 1.5. Directories

Change Directory	cd directory
Make New Directory	mkdir directory
Remove Directory	rmdir directory
Print Working (Show Current) Directory	pwd

## 2. Commands

### 2.1. Command-line Special Characters

#### Quotes and Escape

Join Words	"..."
Suppress Filename, Variable Substitution	'...'
Escape Character	\

#### Separation, Continuation

Command Separation	;
Command-Line Continuation (at end of line)	\

### 2.2. I/O Redirection and Pipes

Standard Output	>
(overwrite if exists)	>!
Appending to Standard Output	>>
Standard Input	<
Standard Error and Output	>&
Standard Error Separately	>& errorfile

( *command* > *output* ) >& *errorfile*

Pipes/ Pipelines *command* | *filter* [ | *filter* ]

#### Filters

Word/Line Count	wc [-l]
Last <i>n</i> Lines	tail [-n]
Sort lines	sort [-n]
Multicolumn Output	pr -t
List Spelling Errors	ispell

### 2.3. Searching with grep

**grep** Command **grep** "pattern" filename  
*command* | **grep** "pattern"

#### Search Patterns

beginning of line	^
end of line	\$
any single character	.
single character in list or range	[...]
character not in list or range	[^...]
zero or more of preceding char. or pattern	*
zero or more of any character	.*
escapes special meaning	\

## 3. C-Shell Features.

### 3.1 History Substitution

Repeat Previous Command	!!
Commands Beginning with <i>str</i>	! <i>str</i>
Commands Containing <i>str</i>	!? <i>str</i> [?]
All Arguments to Prev. Command	!*

#### Word Designators

All Arguments	:*
Last Argument	:\$
First Argument	:^
<i>n</i> 'th Argument	:n

Arguments *x* Through *y* `:x-y`

### Modifiers

Print Command Line `:p`

Substitute Command Line `:[g]s/l/r/`

## 3.2 Aliases

`alias` Command `alias name 'definition'`

*definition* can contain escaped history substitution event and

word designators as placeholders for command-line arguments.

## 3.3. Variable Substitution

Creating a Variable `set var`

Assigning a Value `set var = value`

Expressing a Value `$var`

Displaying a Value `echo $var`

*value* is a single word, an expression in quotes, or an expression that results in a single word after variable, filename and command substitution takes place.

Assigning a List `set var = (list)`

*list* is a space-separated list of words, or an expression that results in a space-separated list.

Selecting the *n*'th Item `$var[n]`

Selecting all Items `$var`

Selecting a Range `$var[x-y]`

Item Count  `$#var`

## 3.4 foreach Lists

Start `foreach` Loop `foreach var (list)`

`foreach` prompts for commands to repeat for each item in list (with >), until you type **end**. Within the loop, `$var` stands for the current item in *list*.

## 3.5. Command Substitution

Replace Command with its Output on Command Line ``...``

## 3.6 Job Control

Run Command in the Background `&`

Stop Foreground Job CTRL-Z

List of Background Jobs `jobs`

Bring Job Forward `%[n]`

Resume Job in Background `%[n] &`

## 4. Processes

Listing `ps [-[ef]]`

Terminating `kill [-9] PID`

Timing `time command`

*time* is a number up to 4 digits. *script* is the name of a file containing the command line(s) to perform.

## 5. Users

Seeing Who is Logged In `who`

`w`

Seeing Your User Name `whoami`

## 6. Managing Files

### 6.1. Looking Up Files

Standard Commands `whereis file`

Aliases and Commands `which command`

Describe Command `whatis command`

Searching Out Files `find dir -name name -print`  
*dir* is a directory name within which to search.  
*name* is a filename to search for.

## 6.2. Finding Changes

Comparing Files `diff leftfile rightfile`

`diff` prefixes a less-than (<) to selected lines from *leftfile* and a greater-than (>) to lines from *rightfile*.

## 6.3. Automating Tasks

Create a Makefile `pico Makefile`

A makefile consists of macro definitions and targets.

Test Makefile `make -n [target]`

Run `make` `make [target]`

## 6.4. Managing Disk Usage

Check Quota `quota -v`

Seeing Disk Usage `df`

`du -s`

## 6.5. Combining and Compressing Files

Create a tarfile `tar cf file.tar file1 file2 ... fileN`

`tar` combines files but does not compress

Create a zipfile `zip filename`

Unzip a file `unzip filename`

## 7. Printing

### 7.1 Formatting Output for Printing

Paginate with Page Headers `pr filename`

in *n* columns `pr -n filename`

Format for Laser Printer `tex document`

### 7.2 The Printer Queue

Print a File `lp [-dpr] filename`

`lpcae filename`

Check Quota `lpquot`

List Queue `lpq`

Stop Job `lprm`

## 8. Miscellaneous

### 8.1 Miscellaneous Commands

List Commands for Subject `man -k subject`

Display Current Date and Time `date`

Log off `exit`

Electronic Mail `pine`

Display Documentation `man command`

### 8.2 Control Keys

Abort Program CTRL-C

Backspace (Delete Last Character) CTRL-H

Pause Display on Screen CTRL-S

Resume Display after CTRL-S CTRL-Q

Send Job to Background CTRL-Z followed by **bg**

Is this document clear? Is it missing crucial informat  
Please mail comments to the handout editor, CAE  
1410 Engineering Drive, or to: editor@engr.wisc.ec