

Princeton University  
 COS 217: Introduction to Programming Systems  
 UNIX and bash

Filenames and Directorynames	
<i>/dir1/.../dirN</i>	Absolute dname
<i>dir1/.../dirN</i>	Relative dname
<i>/dir1/.../file</i>	Absolute fname
<i>dir1/.../file</i>	Relative fname

Special Filename and Directoryname Characters	
<i>fnameord*name</i>	* matches 0 or more characters
<i>fnameord?name</i>	? matches any single character
<i>“fname or dname”</i>	“ allows whitespace in a dname or fname
<i>‘fname or dname’</i>	‘ allows whitespace in a dname or fname
<i>fnameord\name</i>	Backslash (escape) character allows special characters in a dname or fname
<i>~loginid</i>	Home directory of <i>loginid</i>
<i>..</i>	Parent of current directory
<i>.</i>	Current directory

Special Command Characters	
<i>command &lt; fname</i>	Redirect stdin to <i>fname</i>
<i>command &gt; fname</i>	Redirect stdout to <i>fname</i>
<i>command 2&gt; fname</i>	Redirect stderr to <i>fname</i>
<i>command &gt; fname 2&gt;&amp;1</i>	Redirect stdout and stderr to <i>fname</i>
<i>command1   command2</i>	Pipe from <i>command1</i> to <i>command2</i>
<i>^d</i>	End of file
<i>command &amp;</i>	Run <i>command</i> as a background process
<i>^z</i>	Turn my foreground process into a stopped background process
<i>^c</i>	Send a SIGINT signal
<i>↑</i>	Scroll backward through the command history list
<i>↓</i>	Scroll forward through the command history list
<i>!prefix</i>	Reissue the most recently issued command that begins with <i>prefix</i>
<i>!commandnum</i>	Reissue the command whose number is <i>commandnum</i> (see the “history” command)

## Commands

Commands marked with “(bash)” are shell built-in commands.

Commands marked with “(bin)” are executable binary files.

Directory-Related Commands	
pwd	(bash, bin) Print the name of the current directory to stdout
ls [-Fla] [ <i>dname</i> ]	(bin) List the contents of <i>dname</i> to stdout
ls [-Fla] [ <i>fname</i> ]	(bin) List the attributes of <i>fname</i> to stdout
mkdir <i>dname</i>	(bin) Create <i>dname</i>
rmdir <i>dname</i>	(bin) Destroy the empty directory <i>dname</i>
cd [ <i>dname</i> ]	(bash) Make <i>dname</i> the current directory
pushd [ <i>dname</i> ]	(bash) Push the current directory onto the directory stack, and make <i>dname</i> the current directory
popd	(bash) Pop the directory stack, and make the popped directory the current one
dirs	(bash) Print the directory stack to stdout

File-Related Commands	
cat	(bin) Concatenate (print) stdin to stdout
cat <i>fname</i> ...	(bin) Concatenate (print) <i>fname</i> ... to stdout
more	(bin) Print stdin to stdout one screen at a time
more <i>fname</i> ...	(bin) Print <i>fname</i> ... to stdout one screen at a time
cp [-i] <i>sourcefname</i> <i>targetfname</i>	(bin) Copy <i>sourcefname</i> to <i>targetfname</i>
cp [-i] <i>sourcefname</i> <i>targetdname</i>	(bin) Copy <i>sourcefname</i> to <i>targetdname</i>
cp -r <i>sourcedname</i> <i>targetdname</i>	(bin) Copy (recursively) <i>sourcedname</i> to <i>targetdname</i>
mv [-i] <i>sourcefname</i> <i>targetfname</i>	(bin) Rename <i>sourcefname</i> to <i>targetfname</i>
mv [-i] <i>sourcefname</i> ... <i>targetdname</i>	(bin) Move <i>sourcefname</i> ... to <i>targetdname</i>
rm [-i] <i>fname</i> ...	(bin) Remove <i>fname</i> ...
rm -r [-i] <i>dname</i> [ <i>fname</i> ...]	(bin) Remove <i>dname</i> (recursively) and <i>fname</i> ...

Process Control Commands	
jobs	(bash) List the names and jobnums of my background processes to stdout
fg [% <i>jobnum</i> ]	(bash) Move my background process with the given <i>jobnum</i> to the foreground
bg [% <i>jobnum</i> ]	(bash) Turn my stopped background process into a running background process
kill [- <i>signal</i> ] % <i>jobnum</i>	(bash) Send <i>signal</i> to my background process with the given <i>jobnum</i>
ps	(bin) Display a list of my processes
kill [- <i>signal</i> ] <i>pid</i>	(bin) Send <i>signal</i> to the process whose id is <i>pid</i>
exit	(bash) Exit bash
logout	(bash) Exit bash and the terminal session

Configuration Commands	
source <i>fname</i>	(bash) Execute the shell script in <i>fname</i>
export <i>variable</i> = <i>value</i>	(bash) Set environment <i>variable</i> to <i>value</i>
export PATH= <i>dname1</i> : <i>dname2</i> :...	(bash) Set the PATH environment variable indicating that bash should search <i>dname1</i> , <i>dname2</i> , ... to find commands that are specified as relative fnames

export MANPATH= <i>dname1:dname2:...</i>	(bash) Set the MANPATH environment variable indicating that the man command should search <i>dname1</i> , <i>dname2</i> , ... to find man pages
<i>variable=value</i>	(bash) Set shell <i>variable</i> to <i>value</i>
PS1="\! \h--> "	(bash) Set the PS1 shell variable to indicate that the command prompt should contain the history number of the next command and the computer name
set -o <i>shelloption</i>	(bash) Turn on <i>shelloption</i>
set +o <i>shelloption</i>	(bash) Turn off <i>shelloption</i>
set -o ignoreeof	(bash) Turn on the ignoreeof shell option to indicate that ^D entered at the bash prompt should not terminate bash
set -o noclobber	(bash) Turn on the noclobber shell option to indicate that bash should not overwrite files via redirection
alias <i>aliasname=string</i>	(bash) Create an alias definition such that <i>aliasname</i> as an abbreviation for <i>string</i>
unalias <i>aliasname</i>	(bash) Destroy the alias definition that defines <i>aliasname</i>

### Miscellaneous Commands

history	(bash) Print a command history list to stdout
passwd <i>oldpassword</i>	(bin) Change my password from <i>oldpassword</i>
wc [ <i>fname ...</i> ]	(bin) Print a count of characters, words, and lines in <i>fname ...</i>
date	(bin) Print the date and time to stdout
printenv	(bin) Print all environment variables to stdout
echo [ <i>arg ...</i> ]	(bash, bin) Print <i>arg ...</i> to stdout
chmod <i>mask fnameordname ...</i>	(bin) Set the permissions of <i>fnameordname ...</i> as indicated by <i>mask</i>
umask <i>mask</i>	(bash, bin) Set the default permissions used when creating new files and directories as indicated by <i>mask</i>
who	(bin) Print information about current users to stdout
grep <i>string fname ...</i>	(bin) Print each line of <i>fname</i> that contains <i>string</i> to stdout
sort	(bin) Print each line of stdin in lexicographic order to stdout
diff <i>fname1 fname2</i>	(bin) Print an indication of the differences between the contents of <i>fname1</i> and <i>fname2</i> to stdout
which <i>command</i>	(bin) Search PATH for <i>command</i> , and print the <i>dname</i> where it was found to stdout
finger <i>loginid</i>	(bin) Print information about user <i>loginid</i> to stdout

### Software Development Commands (described later in the course)

xemacs	(bin) Create or edit a text file using the Emacs editor
gcc	(bin) Preprocess, compile, assemble, and link a program
gdb	(bin) Debug a program
make	(bin) Build a program
ar	(bin) Create an archive file containing object code
rsc, ci, co	(bin) Use the rcs source code control system
gprof	(bin) Analyze the performance of a program

### Commands for Getting Help

man [-s <i>sectionnum</i> ] <i>pagename</i>	(bin) Print to stdout the UNIX manual page (from <i>sectionnum</i> ) whose name is <i>pagename</i>
apropos <i>keyword</i>	(bin) Print to stdout each UNIX manual page NAME line that contains <i>keyword</i>