NL Description	Bash Command	Time (s)
Make "bar" executable	chmod +x bar	4.0
list PID of a group leader	jobs -lp	4.0
Save the system host name in variable "HOSTNAME"	HOSTNAME="`hostname`"	4.0
Sort file "foo.txt" by line to standard output	sort foo.txt	4.0
Save the user name of the current user to variable "whoami"	whoami=$(whoami)	5.0
Find all files/directories under current directory and sort them	find | sort	5.0
Report file system containing path to the current working directory inodes usage.	df -i $PWD	5.0
List files and directories one level deep in the current directory tree	tree -L 2	5.0
change group of the file myfile to group friends	chown :friends myfile	5.0
Display the last slash-separated part of path, in this case "example".	basename /usr/local/svn/repos/example	5.0
Print linux group names on multiple lines instead of single line output	groups | tr \ \\n	5.0
Find all files/directories named 'text' under current directory	find -name "text"	5.0
Print out the full path name of "mypathname" with dots resolved	readlink -ev mypathname	6.0
find all regular files which have been modified in the last 48 hours in home folder	find ~ -type f -mtime -2	8.0
Find directories modified within the last 7 days	find . -mtime -7 -type d	8.0
Search for " 840" in history	history | grep " 840"	8.0
Creates temporary folder in TMPDIR (if defined) or in '/tmp/', and stores path to created folder in 'tmpdir' variable.	tmpdir=$(mktemp -d)	8.0
Find all files and directories and count them	find ./ | wc -l	8.0
Overwirte file '/path/to/your/file' with random content, showing progress while execution.	shred -v -n 1 /path/to/your/file \#overwriting with random data	8.0
Find all of the hidden files in the current directory and its sub-directories.	find . -type f -name ".*"	8.0
List all environment variables (name and value) whose name either equals HOME or PATH, or starts with GO	env | grep '^\(GO\\|HOME=\\|PATH=\)'	8.0
Create intermediate directoriy path2 as required and directories a..z	mkdir -p path2/\{a..z\}	8.0
List all files in /home/bozo/projects directory tree that were modified exactly one day ago	find /home/bozo/projects -mtime 1	8.0
find md5sum of a string stored in $pass	printf '\%s' "$pass" | md5	8.0
ssh into "ssh.myhost.net" as user "myusername" and run command "mkdir -p $2"	ssh myusername@ssh.myhost.net "mkdir -p $2"	10.0
Composes full process tree with process id numbers, and prints only those strings that contain 'git'.	pstree -p | grep git	10.0
Print the IP addresses for the current host name	hostname -I | awk -F" " '\{print $1\}'	10.0
List directories in the current working directory and remove the trailing "/"	ls -d */|sed 's|[/]||g'	10.0
Remove the last two alphanumeric file extensions from "mpc-1.0.1.tar.gz"	echo "mpc-1.0.1.tar.gz" | sed -r 's/\.[[:alnum:]]+\.[[:alnum:]]+$//'	10.0
Lists all subdirectories in a current folder, removing trailing slash.	ls -d */|sed 's|[/]||g'	10.0
Copy "src/prog.js" and "images/icon.jpg" to "/tmp/package" keeping relative path names	rsync -R src/prog.js images/icon.jpg /tmp/package	10.0
Perform a dry run to recursively copy "test/a" to "test/dest" excluding "test/a/b/c/d"	rsync -nvraL test/a test/dest --exclude=a/b/c/d	10.0
Print the directory name of the full real path to the current script	echo "dirname/readlink: $(dirname $(readlink -f $0))"	10.0
List files in the current directory and below except for GIT files	find . -not -iwholename '*/.git/*'	10.0
Delete all lines in "file" that contain "pattern", using "temp" as temporary working file (ie. shouldn't exist).	grep -v "pattern" file > temp \&\& mv temp file	10.0
Find all the files in file system which are greater than 50MB and less than 100MB	find / -size +50M -size -100M	10.0
display all the files along with their group name in the folder /home which do not belong to the group test	find /home ! -group test -printf "\%p:\%g\n"	12.0
Find all files/directories named 'foo' under current directory tree without descending into directories named 'foo'	find . -name foo -type d -prune -o -name foo	12.0
Exclude directory from find . command	find ./ -path ./beta/* -prune -o -iname example.com -print	12.0
Write contents of "/sys/kernel/debug/tracing/trace_pipe" to standard output and to "tracelog.txt" executing as a root user	sudo cat /sys/kernel/debug/tracing/trace_pipe | tee tracelog.txt	12.0
Find all files/directories with '.in' extension in the directory $directory and its subdirectories	du -a $directory | awk '\{print $2\}' \| grep '\.in$'	12.0
Find all *.txt files except for those on other file systems	find / -name "*.txt" -mount 2> /dev/null	12.0
Remove files that are less than 1MB in size under <directory>	find <directory> -type f -size -1M -delete	12.0
Print the absolute path of third-level files under the current directory tree and number the output	ls -d -1 $PWD/**/*/* | nl	12.0
Find all regular files with '.txt' extension excluding 'README.txt' files under current directory tree	find . -type f -name "*.txt" ! -name README.txt -print	12.0
List all IP addresses assigned to current hostname, pausing for user interaction after each page.	more /etc/hosts | grep '[[:space:]]*'`hostname`'[[:space:]]*' | awk '\{print $1\}'	15.0
search for a word in all c files in the current folder	find . -name '*.c' | xargs grep 'stdlib.h'	15.0
Remove all libGLE* files from the current directory tree	find . -name libGLE* | xargs rm -f	15.0
Counts lines in each of *.php files in a current folder and subfolders and prints total count as well.	find . -name "*.php" | xargs wc -l	15.0
Search all files called "abc" that reside in the current directory tree for string "xyz"	find . -name "abc" -exec grep "xyz" \{\} \;	15.0
Remove recursively Emacs backup files in the current directory	find . -name '*~' -print0 | xargs -0 rm	15.0
Search the current directory tree for filenames matching the pattern '[mM][yY][fF][ii][lL][eE]*'	find . -name '[mM][yY][fF][ii][lL][eE]*'	15.0
Finds strings with dot-separated sequence of numbers, and prints part of that sequence before the second and third dot.	echo "$f" | grep -Eo '[0-9]+[.]+[0-9]+[.]?[0-9]?' | cut -d. -f3	15.0
List all .svn files/directories under current directory	find . -name .svn -exec ls \{\} \;	15.0
List all IP addresses assigned to current hostname, pausing for user interaction after each page.	more /etc/hosts | grep `hostname` | awk '\{print $1\}'	15.0
search for the word text in all the python files in the current folder	find . -iname '*py' -exec grep "text" \{\} \;	15.0
find all files under the current directory that end in "foo" and execute somecommand on each quoted filename	find . -name "*.foo" -exec somecommand "\{\}" \;	15.0
Find all files/directories under $TARGET_DIR directory tree matching the posix extended regular expression \".*/$now.*\" (where $now is a variable) and save the output in file $FILE_LIST	find $TARGET_DIR -regextype posix-extended -regex \".*/$now.*\" -fprint $FILE_LIST	15.0
Search the entire file hierarchy for files larger than 100 megabytes and delete them.	find / -size +100M -exec /bin/rm \{\} \;	15.0
reverse both words and lines in file	tac filename | perl -lane 'print join(" ", reverse(@F))'	15.0
Find all directories under current directory and change their permission to 500	find . -type d -exec chmod 500 \{\} \;	15.0
Forcefully delete all files in the current directory that begin with spam-	find . -name 'spam-*' | xargs rm	15.0
List the number of occurrences of each unique character in "The quick brown fox jumps over the lazy dog" sorted from most frequent to least frequent	echo "The quick brown fox jumps over the lazy dog" | grep -o . | sort | uniq -c | sort -nr	15.0
Search the home directory tree for video files	find ~ -type f -regex '.*\.\(mkv\\|mp4\\|wmv\\|flv\\|webm\\|mov\)'	18.0
Counts lines in each *.php file sorted by file path.	find . -name '*.php' -type f | sort | xargs wc -l	18.0
search for the text file "file.txt" and display its parent directory	cd /nfs/office/ \&\& find . -name 'file.txt' | sed -r 's|(\./([^/]*/)*).*|\1|'	20.0
Test if files named 'something' were found in Dir/	[[! -z `find 'Dir/' -name 'something'`]] \&\& echo "found" || echo "not found"	20.0
Prints top-ten list of most used utilities from history.	history | awk '\{ print $2 \}' | sort | uniq -c |sort -rn | head	20.0
change the extension of all the ".lst" files in the current folder to "a.lst"	find -name '*.lst' -exec rename .lst a.lst \{\} \;	20.0
Find the top 5 biggest files	find . -type f -exec ls -s \{\} \; | sort -n -r | head -5	20.0
Compose filepath as folder path where file $SRC is located, and lowercase filename of $SRC file, and save it in 'DST' variable	DST=`dirname "$\{SRC\}"`/`basename "$\{SRC\}" | tr '[A-Z]' '[a-z]'`	20.0
Find all foo.mp4 files in the current directory tree and print the pathnames of their parent directories	find . -name foo.mp4 | sed 's|/[^/]*$||'	20.0
Find all *.jpg files under current directory and print only unique names	find . -name *.jpg -exec basename \{\} \; | uniq -u	20.0
Print the contents of all file* files under current directory with white space safety in file names	find . -name "file*" -print0 | xargs -0 perl -ple ''	20.0
Find all files in /var/www/html/zip/data/*/*/*/*/* that are older than 90 days and print their parent directory paths	find /var/www/html/zip/data/*/*/*/*/* -type f -mtime +90 | sed 's|/[^/]*$||'	20.0
Replace the first occurrence of "string1" on each line with "string2" in all regular files in the current directory tree	find ./ -type f -exec sed -i 's/string1/string2/' \{\} \;	20.0
List the directory paths of all file.ext files under present working directory	find `pwd` -name "file.ext" -exec dirname \{\} \;	20.0
get a PID of a process with name 'test.sh \&'	jobs -l | grep 'test.sh \&' | grep -v grep | awk '\{print $2\}'	20.0
search all html files in the current folder and remove all the empty lines	find ./ -type f -name '*.html' | xargs sed -i '/^$/d'	20.0
Make a .tar.bz2 archive of all *.txt files from the dir/ directory tree	find dir/ -name '*.txt' | tar -c --files-from=- | bzip2 > dir_txt.tar.bz2	25.0
Display the 5 smallest files in the current directory and its sub-directories.	find . -type f -exec ls -s \{\} \; | sort -n | head -5	25.0
Search the current directory tree recursively for files containing "needle text"	find . -type f -print0 | xargs -0 grep -IZl . | xargs -0 grep "needle text"	25.0
Find all files under and below the current working directory with the word California in the file (case insensitive), and count the number of lines in the output	find . -type f -exec grep -i California \{\} \; -print | wc -l	25.0
List files greater than 1024KB under /path/to/directory and print the time and size on stdout	find /path/to/directory -type f -size +1024k -exec ls -lh \{\} \; | awk '\{ print $8 ": " $5 \}'	25.0
alias restart_rails for command 'kill -9 `cat tmp/pids/server.pid`; rails server -d'	alias restart_rails='kill -9 `cat tmp/pids/server.pid`; rails server -d'	25.0
Clean up all zombie processes by sending SIGTERM signal to their parent process, which requests them to terminate.	kill $(ps -A -ostat,ppid | awk '/[zZ]/\{print $2\}')	25.0
Save the number of records in the system hostname that contain numbers in variable "server_id"	server_id=`hostname | tr 'A-Za-z-.' ' ' | tr -d '[[:space:]]' | awk '\{print NR\}'`	25.0
In a ssh session, set the variable 'user' to the last dot-separated number of the client's IP address.	export user=`env|grep -i SSH\_CLIENT|cut -d' ' -f1|cut -d'.' -f4`	25.0
Search the current directory tree for all image files	find . -type f -regex ".*\.\(jpg\\|jpeg\\|gif\\|png\\|JPG\\|JPEG\\|GIF\\|PNG\)"	25.0
Find all files under $\{searchpath\} that match the regex '"$\{string1\}".*"$\{string2\}".*"$\{string3\}"' ($\{string1\} ... won't be expanded) in their contents	find `echo "$\{searchpath\}"` -type f -print0 \| xargs -0 grep -l -E '"$\{string1\}".*"$\{string2\}".*"$\{string3\}"'	25.0
Finds string with text "string to be searched" in any cased files like "*.cs" recursively in a current folder.	find ./ -type f -iname "*.cs" -print0 | xargs -0 grep "content pattern"	25.0
Move all *.pdf.marker files and their corresponding *.pdf files under $\{INPUT\} to $\{OUTPUT\}	find $\{INPUT\}/ -name "*.pdf" -exec mv '\{\}' '\{\}'.marker $\{OUTPUT\} \;	25.0
Saves date of the first Sunday in month $mo of year $yo in the 'do' variable.	do=$(cal -m $mo $yo|awk 'NR>2\&\&!/^/\{print$1;exit\}')	30.0
Sort all directory names matching folder_* and go to the last one.	cd $(find . -maxdepth 1 -type d -name "folder_*" | sort -t_ -k2 -n -r | head -1)	30.0
search for all the files excluding directories ending with old or sccs or core or orig in the project folder and save the output to the file exclude	find project ! -type d -print | egrep '/,|\%$\|~$|\.old$\|SCCS\|/core$|\.o$\|\.orig$' > Exclude	30.0
Print DISPLAY of "orschiro" user	who | awk -F '[()]' '/orschiro/\{print $(NF-1)\}' | grep -v orschiro | uniq	30.0
Cut all remote paths from HTTP URLs received from standard input (one per line) keeping only the protocol identifier and host name, of the form http://example.com	sed -n 's;\(http://[^/]*\)/.*;\1;p'	30.0
Search my_folder recursively for text files containing "needle text"	find my_folder -type f -exec grep -l "needle text" \{\} \; -exec file \{\} \; | grep text	30.0
set alias "killaf" for command "kill -9 `psu|grep MF1pp|grep -v grep|awk '\{print $2\}'`"	alias killaf="kill -9 `psu|grep MF1pp|grep -v grep|awk '\{print $2\}'`"	30.0
Print newline, word and byte count for all .h, .c, .cpp, .php and .cc files under current directory tree and also show the total counts	wc `find . -name '*.[h|c|cpp|php|cc]'`	30.0
find all normal/regular files in the folder "//path/to/source/Directory" and calculate the md5sum of them and save the output to file Output.txt	find //path/to/source/Directory -type f -exec md5sum \{\} + | awk '\{print $0\}' > Output.txt	30.0
Cut all remote paths from HTTP URLs received from standard input (one per line) keeping only the protocol identifier, host name, and trailing slash, of the form http://example.com/	sed -n 's;\(http://[^/]*/\).*;\1;p'	30.0
Count the number of files in the /usr/ports directory tree whose names begin with 'pkg-plist' and which contain 'dirrmtry'	find /usr/ports/ -name pkg-plist* -exec grep dirrmtry '\{\}' '+' | wc -l	30.0
Move *wp-admin/index.php files under /var/www/ to ./index_disabled	find /var/www/ -path '*wp-admin/index.php' -exec mv \{\} $(dirname \{\})/index_disabled	35.0
Scan every file in /etc for IPV4 addresses.	find /etc -exec grep '[0-9][0-9]*[.][0-9][0-9]*[.][0-9][0-9]*[.][0-9][0-9]*' \{\} \;	40.0
display all the ip addresses in all the files that are present in /etc folder	find /etc -exec grep '[0-9][0-9]*[.][0-9][0-9]*[.][0-9][0-9]*[.][0-9][0-9]*' \{\} \;	40.0
find all files under the /etc directory and display any IP address patterns in them	find /etc -exec grep '[0-9][0-9]*[.][0-9][0-9]*[.][0-9][0-9]*[.][0-9][0-9]*' \{\} \;	40.0
Search all files under and below /etc for IP addresses	find /etc -exec grep '[0-9][0-9]*[.][0-9][0-9]*[.][0-9][0-9]*[.][0-9][0-9]*' \{\} \;	40.0
Find all IP addresses in /etc directory files	find /etc -exec grep '[0-9][0-9]*[.][0-9][0-9]*[.][0-9][0-9]*[.][0-9][0-9]*' \{\} \;	40.0
Gets IP address of $\{NET_IF\} network interface.	NET_IP=`ifconfig $\{NET\_IF\} | grep -Eo 'inet (addr:)?([0-9]*\.)\{3\}[0-9]*' | grep -Eo '([0-9]*\.)\{3\}[0-9]*' | grep -v '127.0.0.1'`	40.0
Search for the regex '[0-9][0-9]*[.][0-9][0-9]*[.][0-9][0-9]*[.][0-9][0-9]*' in all files under /etc	find /etc -exec grep '[0-9][0-9]*[.][0-9][0-9]*[.][0-9][0-9]*[.][0-9][0-9]*' \{\} \;	40.0
Remove trailing white spaces from all files under current directory ignoring .git and .svn directories	find . -not \(-name .svn -prune -o -name .git -prune \) -type f -print0 | xargs -0 sed -i '' -E "s/[[:space:]]*$//"	40.0
ssh into desired vagrant host "HOST" without running "vagrant ssh"	ssh $(vagrant ssh-config HOST \| sed '/^[[:space:]]*$/d' | awk 'NR>1 \{print " -o "$1"="$2\}') localhost	40.0
Remove all files matching the pattern *[+\{;"\\=?~()<>\&*|$]* under current directory	find . -name '*[+\{;"\\=?~()<>\&*|$]*' -exec rm -f '\{\}' \;	40.0
Gets IP address of only primary network interface.	ifconfig `ip route | grep default | head -1 | sed 's/\(.*dev \)\([a-z0-9]*\)\(.*\)/\2/g'` | grep -oE "\b([0-9]\{1,3\}\.)\{3\}[0-9]\{1,3\}\b" | head -1	45.0
Print the lines in "table.txt" grouped by the first field	cat table.txt | awk '\{print $1\}' | sort | uniq | xargs -i grep \{\} table.txt	45.0
Run an awk program on every TXT file found in the current directory tree	find . -name "*.txt" -print -exec awk '\{if($9!=""\&\&n<11)\{print;n++\}\}' \{\} \;	45.0
Write the last line of "$file" to standard output and remove it from "$file"	tail -n 1 "$file" \| tee >(wc -c \| xargs -I \{\} truncate "$file" -s -\{\})	45.0
Calculate MD5 sums for all regular files in the current directory tree	find . -type f | while read f; do g=`md5sum $f | awk '\{print $1\}'`; echo "$g $f"> $f-$g.md5; done	50.0
display the contents of all the regular files in a folder in the sorted order of their depth	cat $(find src/js -type f -name "*.js" -printf "\%d\t\%p\n" | sort -n | sed -e "s|[0-9]*\t||")>min.js	60.0
Print URL "http://www.blabla.bla/forum-detail/?ft=72260\&fid=34\&\&pgr=" followed by a number ranging from 1 to 786	yes 'http://www.blabla.bla/forum-detail/?ft=72260\&fid=34\&\&pgr=' | nl -ba | sed 786q | grep . | awk '\{print $2$1\}'	60.0
Check if command "c++" and command "g++" are equal	[`md5sum $(which c++) | cut -d' ' -f1` == `md5sum $(which g++) | cut -d' ' -f1`] \&\& echo Yes, equal content || echo No, unequal content	60.0
display all html files in current folder and replace some pattern in all these files	find -name "*.htm" | while read file; do sed "s|<title>sometext</title>|<title>$\{file\#\#*/\}</title>|g" -i $file; done	60.0
find all the png files in current folder which are present in the pattern list file "search.txt" and copy them to another directory	find . -name '*.png' | grep -f <(sed s?^?/[0-9]_[0-9]_[0-9]_? search.txt) | xargs -i\{\} cp \{\} /path/to/dir	60.0
Find recursively the latest modified file in the current directory	find . -type f | sed 's/.*/"\&"/' | xargs ls -E | awk '\{ print $6," ",$7 \}' | sort | tail -1	60.0
Find the largest original ".gz" file in the current directory tree	find . -name '*.gz' -print | xargs gzip -l | awk '\{ print $2, $4 ;\}' \| grep -v '(totals)$' | sort -n | tail -1	70.0
Print linker search path using gcc formatted on new lines	gcc -print-search-dirs | sed '/^lib/b 1;d;:1;s,/[^/.][^/]*/\.\./,/,;t 1;s,:[^=]*=,:;,;s,;,; ,g' | tr \; \\012	75.0
Prints last modified file in a current folder with modification time.	find . -type f | sed 's/.*/"\&"/' | xargs ls -E | awk '\{ print $6," ",$7 \}' | sort | tail -1	75.0
Print characters in variable "$a" that exist in variable "$b"	echo "$(comm -12 <(echo "$a" | fold -w1 | sort | uniq) <(echo "$b" | fold -w1 | sort | uniq) | tr -d '\n')"	90.0
Subtract each column in File2 from matching column in File1, output the result in a similarly formatted table	awk 'FNR==NR \{ for(i=2;i<=NF;i++) a[$1][i]=$i; next \} \{ for(j=2;j<=NF;j++) $j-=a[$1][j] \}1' File2 File1 | rev | column -t | rev	120.0
Find all files/directories under /path/to/files/* paths and print the timestamp in YmdHMS format along with their paths and object of symlinks, stat them and apply the sed replacement "s/(.*\)\([0-9]\{4\}\)-([0-9]\{2\}\)-([0-9]\{2\}) ([0- …	find /path/to/files/* -printf "\%TY\%Tm\%Td\%TH\%TM\%TS|\%p|\%l\n" -exec stat -Lc "|\%y" \{\} \; | sed -r "s/\(.*\)\([0-9]\{4\}\)-\([0-9]\{2\}\)-\([0-9]\{2\}\) \([0- …	120.0
Delete all shared memory and semaphores for the current user on linux	ipcs -a | nawk -v u=`whoami` '$5==u \&\&(($1=="m" \&\& $9==0)||($1=="s"))\{print "ipcrm -"$1,$2,";"\}' | /bin/sh	150.0
Convert Unix `cal` output to latex table code.	cal | sed '1d;2/\{h;s/./ /g;x\};/^\s*$/b;G;s/\n/;/;s/^...\(.\)\(.\{15\}\).*/\1/;s/.../ …	180.0
List every directory under current directory with their child executable files	find `pwd` -perm 111 -type f | sort -r | xargs -n1 -I\{\} sh -c "dirname \{\};basename \{\}" | awk '/^\// \{dir=$0 ; if (dir != lastdir) \{print;lastdir=dir\}\} /^[^\/]/\{print\}'	180.0
Convert Unix `cal` output to latex table code.	cal 02 2012|perl -lnE'$.==1\|\|eof\|\|do\{$,="\&";$\="\\\\\n";$l=$_;print map\{substr($l,$_*3,3)\}(1..5)\}'	200.0
Convert Unix `cal` output to latex table code.	cal 02 2012|perl -F'(.\{1,3\})' -anE'BEGIN\{$,="\t\&";$\="\\\\\n"\}$.==1\|\|eof\|\|do\{$i//=@F;print@F[map\{$_*2-1\}(1..$i/2)]\}'	240.0