COHERENT™

A Multi-User, Multi-Tasking Operating System for the IBM-PC/AT and Compatible 286 or 386 Based Computers.

Pocket Reference Guide

ac — Summarize login accounting information
ac [**-dp**] [**-w** *wfile*][*username ...*]
Options:
-d	Itemize for each midnight-midnight period
-p	Itemize by individual users
-w *wtmp*	Obtain raw statistics from *wtmp* rather than **/usr/adm/wtmp**

If users are specified, only they are considered.

accton — Enable/disable process accounting
/etc/accton [*file*]
Default *file* is **/usr/adm/acct**. If no *file* argument is used, disable process accounting.

alias — Set an alias
alias [*name***[=***value ...***]]**
ksh only.

almanac — Print an almanac entry for this date
almanac [*month day***]**

apropos — Find manual pages on a given topic
apropos *topic* **[** *topic* **...]**

ar — The librarian/archiver
ar *option* **[***modifier***][***position***]** *archive* **[***member ...***]**
Options:
d	Delete given members
m	Move member to indicated position (default, end)
p	Print members
q	Quick append, put members at end with no checking
r	Replace each member specified in the archive
t	Print a table of members (default, all)
x	Extract the specified members (default, all)

Modifiers:
a	Place new member after *position* in archive
b	Place new member before *position* in archive
c	Suppress message when new archive is created
i	Insert new member before *position* in archive
k	Preserve modify time of file (with options **r**, **q**, or **x**)
l	Use current directory for temporaries (default, **/tmp**)
s	Update ranlib header even if not present (with options **r** or **m**)
u	Update: replace members only if newer than those in archive
v	Print extra information when used with certain options

as — i80386 assembler
as [-o *outfile*] **[-bfglnpwxX]** *infile*
Options:
-D*name***=***string*	Initialize string variable *name* to *string*.
-E*name***=***value*	Initialize numeric variable *name* to *value*.
-a	Set alignment for data objects
-b	Reverse bracket sense; that is, use **()** for expressions and **[]** for code.
-f	Reverse order of operands from assembler form to that of Intel documentation or 80286 version of **as**.
-g	Make undefined symbols **.globl**.
-l	Generate output listing.
-n	Turn off the insertion of **nop**s to correct 80386 errors.
-o *outfile*	Name the file into which the relocatable object is written
-p	Don't use '%' on register names; e.g., use **ax**, not **%ax**.
-Q	Quiet: Suppress all error messages, no matter how awful an error they indicate
-w	Disable warning messages.
-x	Remove all non-global symbols from the common symbol output.
-X	Remove all non-global symbols starting with **.L** from the common symbol output.

Options and file names can be interspersed on the command line.

as assembles files written in the 80286 dialect of **as** or -style assembly language. It generates relocatable object modules that can be linked with objects compiled by the C compiler. It also contains a number of features not available with the 80286 dialect of **as**, including macro assembly.

asfix — Convert assembly-language programs into 80386 format
asfix < *oldfile* **>** *newfile*

asymkdev — Create nodes for asynchronous devices
/conf/asymkdev [-u] [*async_file* [*outfile*]]

asypatch — Patch a kernel file for an asynchronous configuration
/conf/asypatch [-v] <​*kernel_name*​**> <***async_file*

at — Execute commands at given time
at [**-v**] [**-c** *command*] *time* [[*day*] *week*] [*file*]
at [**-v**] [**-c** *command*] *time month day* [*file*]
Options:
   -c   Following argument gives command
   -v   Print time for which command is set
If *file* is given, read commands from it. If neither *file* nor **-c** is given, read commands from stdin.

ATclock — Read or set the AT realtime clock
/etc/ATclock *[yy[mm[dd[hh[mm[.ss]]]]]]*
With an argument, **ATclock** sets your computer's realtime clock. With no argument, it reads it.

awk — Pattern-scanning language
awk [*POSIX or GNU style options* **] -f** *program-file* **[--]** *file ...*
awk [*POSIX or GNU style options* **] [--]** *program-text file ...*

bad — Maintain list of bad blocks
bad [-acdl] *device* **[** *block ...* **]**
Options:
   a   Add blocks
   c   Clear bad-block list
   d   Delete blocks
   l   List blocks

badscan — Build bad block list
/etc/badscan [-v] [-o *proto* **] [-b** *boot* **]** *device size*
/etc/badscan [-v] [-o *proto* **] [-b** *boot* **]** *device xdevice*
Options:
   -b *boot*   Insert bootstrap *boot* into *proto*
   -o *proto*   Write prototype into file *proto*
   -v   Print estimate of time remaining
Scan *device* of *size* bytes (or size given in hard disk partition table *xdevice*) for bad blocks, write prototype to stdout.

banner — Print large letters
banner [*argument ...*]
Print each *argument* as one line of large-text output. If no arguments, print each line from stdin as a line of large output.

basename — Strip path information from a file name
basename *file* [*suffix*]

bc — Interactive calculator with arbitrary precision
bc [-l] [*file ...*]
Option:
   -l   Use the extended **bc** library
If no *file* is specified, **bc** reads stdin.

break — Exit from shell construct
break [*n*]
Exit from *n* (default, one) **for**, **until**, or **while** constructs. The shell executes **break** directly.

build — Install onto a hard disk
/etc/build

builtin — Execute a command as a built-in command
builtin *command* **[** *arg ...* **]**
ksh only.

c — Print multi-column output
c [-1*N* **] [-w***N* **] [-012]**
Options:
   -1*N*   Set the page length to *N* lines
   -w*N*   Set the page width to *N* columns
   -0   Order fields horizontally across the page
   -1   Order fields vertically down each column (default)
   -2   Special case of -1

cal — Print a calendar
cal [*month*] [*year*]

calendar — Reminder service
calendar [-a] [-f*file* **]… [-d[***date***]] [-w[***date***]] [-m[***month***]]**
Options:
   -a   Search calendars of all users and send mail
   -f*file*   Search each *file* in order given
   -d[*date***]**   Print all entries matching *date*
   -w[*date***]**   Print entries in the week beginning with *date*
   -m[*month***]**   Print entries in the given *month*
The default calendar is **$HOME/.calendar**. The default date is today.

cancel — Cancel a print job
cancel [*job* **[** *... job***]] [-all]**

captoinfo — Convert termcap data to terminfo form
captoinfo [*filename***]**

case — Execute commands conditionally according to pattern
case *token* **in [***pattern* **[|** *pattern***] …)** *sequence* **;;] … esac**
The shell executes **case** directly.

cat — Concatenate the contents of a file to the standard output
cat [**-u**][*file ...*]
Option:
   -u   Do not buffer output in 512-byte blocks
File '-' indicates the standard input. If no *file* is specified, **cat** reads stdin.

cc — C compiler
cc [*compiler options***]** *file . . . .* **[***linker options***]**
Options:
   -A

                   option: Invoke automatically should any errors be
                  found during compilation, to allow user to edit source
                  and then recompile
   -a   Create an executable file named **a.out**, unless option **-o**
                  is used explicitly
   -B   Backup option: Use alternate versions of various
                  compiler phases
   -c   Compile only — no load
   -D*name***[=***value***]**Tell **cpp** to define *name* with *value*
   -E   Run **cpp** only and send its output to stdout
   -f   Link in routines to format floating-point output
   -g   Generate debugging information
   -I*name*   Tell **cpp** to look for header files in directory *name*
   -K   Keep intermediate files
   -L*directory*   Tell the linker **ld** to search *directory* for its libraries
                  before it searches the directories named in the
                  environmental variable **LIBPATH**
   -l*name*   Pass **/lib/lib***name***.a** to linker **ld**
   -M*string*   Use alternative machine versions
   -O   Run peephole optimizer of C compiler

-q[p01ab2s]	Quit after specified pass
-Q	Quiet: suppress all error messages, no matter how awful an error they indicate
-S	Place C compiler assembler output in a **.s** file
-T*size*	Tell **cc** to use buffers of *sizen* bytes each, instead of its default 64-kilobytes buffers. **cc** reads the environmental variable **TMPDIR** to find where you want it to write its temporary files.
-t[p01ab2sdlrt]	Take specified compiler phases.
-U*name*	Tell **cpp** to remove any initial definition of *name*
-V	Run verbosely
-V*name*	Toggle variant **V***name*

Compiles files ending **.c**; assembles files ending in **.s**; passes other options and files to the linker **ld**.

cd — Change directory
cd *directory*
If no *directory* specified, **$HOME** is assumed. The shell executes **cd** directly.

cdmp — Dump COFF files into a readable form
cdmp [-adlrs] *filename*

Options:

-a	Suppress auxiliary symbol entries
-d	Suppress data dumps
-l	Suppress line numbers
-r	Suppress relocation entries
-s	Suppress symbol entries

cdplayer — Play audio CDs
cdplayer [eject info pause play [*track***] resume skip stop volume** *level*]

cdv — Interface to CD-ROM devices
cdv [*directory***]**

cdview — Read a file from a CD-ROM
cdview [*file***]**

cgrep — Pattern search for C source programs
cgrep [-clnsA] [-r *new*] *expression file* **...**

Options:

-A	Build *error list* for interactive editing using , like **-A** option to the **cc** command.
-c	Print all C comments
-l	Return file where *expression* found
-n	Prefix each line containing *expressions* with its number in its source file
-r	Replaces each *expression* with *new*
-s	Print all C strings

chase — Highly amusing video game
/usr/games/chase [-c] [*speed* **]**

Options:

-c	Color video card
speed	Speed of game: the lower the number, the faster the speed (default, 10)

check — Check file system
check [-s] *filesystem* ...

Option:

-s	Salvage as much as possible, given the problems detected

checkerr — Check the mail system for errors
/usr/lib/mail/checkerr

chgrp — Change the group owner of a file
chgrp *group file ...*

chmod — Change the modes of a file
chmod **+***modes file*
chmod **-***modes file*
Mode may be octal or a comma-separated symbolic list:
[*which***]***howperm***...[,...]**
which:

a	User, group, and other permissions
g	Group permissions
o	Other permissions
u	User permissions

Missing *which* implies that 'a', 'g', 'o', and 'u' can be combined.
how:

=	Set permissions
+	Add permissions
-	Take away permissions

perm:

g	Current group permissions
o	Current other permissions
r	Read
s	Setuid on execution
t	Sticky bit (save text)
u	Current user permissions
w	Write
x	Execute

chmog — Change mode, owner, and group simultaneously
chmog *mod own grp file ...*

chown — Change the owner of files
chown *owner file ...*

chreq — Change priority, lifetime, or printer for a job
chreq [-d*printer***] [-l***lifetime***] [-p***priority***]** *job*

Options:

-d*printer*	Move *job* to the queue for *printer*
-l*lifetime*	Change the lifetime of *job*, where *lifetime* is one of **T** (temporary), **S** (short-term), or **L** (long-term)
-p*priority*	Change priority of *job* to *priority*, which is one of **0** (highest priority) to **9** (lowest priority)

chroot — Change root directory
chroot *directory program ...*

ckermit — Interactive inter-system communication and file transfer
ckermit [-abcdefghiklpqrstwx] [*file ... * **]**

Options

-a *filename*	Give an alternate name to file
-b *baudrate*	Set the baud rate of the device to *baudrate*
-c	Connect
-d	Use debug mode
-e *n*	Set the length of the packet to *n*
-f	Send a "finish" command to a remote server
-g *file*	Ask a remote system to send *file*
-h	Print help
-i	Specify that the file being transferred is binary
-k	Passively receive files
-l *device*	Name the serial device to be used
-n	Like **-c**, but used after a protocol transaction has occurred
-p *x*	Set parity to *x*
-q	Quiet mode — no messages
-r	Receive files
-s *file*	Send *file*
-t	Specify half duplex
-w	Write-protect — avoid file-name collisions for incoming files
-x	Begin server operation

clear — Clear the screen
clear

clri — Clear i-node
/etc/clri *filesystem inumber ...*

cmp — Compare bytes of two files
cmp [-ls] *file1 file2* **[***skip1 skip2***]**
Options:
   -l   Print byte number and bytes at each difference
   -s   Return status (print nothing)
If *file1* is '-', use stdin. If *skip1* and *skip2* are present, they are the number
of bytes to skip before comparing *file1* and *file2*, respectively.

coh_intro — Tour the file system
/etc/coh_intro [> *outfile* **]**

cohtune — Set a variable within a device driver
cohtune *driver tagfield* **"***tagfield* **=** *value***"**

col — Remove reverse and half-line motions
col [**-bdfx**][**-p***n*]
Options:
   -b   Output device cannot backspace
   -d   Double spaced output
   -f   The output device can handle half lines (has precedence
                     over **-d**)
   -p*n*   Set page buffer to *n* lines (default, 128)
   -x   Suppress conversion of white space to tabs on output

comm — Print common lines
comm [-123] *file1 file2*
Options:
   -1   Suppress column 1
   -2   Suppress column 2
   -3   Suppress column 3
Column 1 has lines unique to *file1*; column 2 has lines unique to *file2*;
column 3 has lines common to both files. Both files should be sorted.

compress — Compress a file
compress [-dfvc] [-b*num***] [** *file ...* **]**
Options:
   -b*num*   Set compression to *num*
   -c   Send output to **stdout**
   -d   Decompress, rather than compress
   -f   Force output file, even if no space saved by compression
   -v   Verbose mode

continue — Terminate current iteration of shell construct
continue [*n*]
Terminate current iteration of *n* (default, one) **for**, **until**, or **while**
constructs. The shell executes **continue** directly.

conv — Numeric base converter
conv [*number*]
If no *number* is given, reads one number per line from stdin.

cp — Copy a file
cp [-d] *oldname newname*
cp [-d] *file1 ... fileN directory*
Option:
   -d   Preserve date (*mtime*) on destination files.

cpdir — Copy directory hierarchy
cpdir [*option ...*] *dir1 dir2*
Options:
   -a   Verbose file by file account on one line
   -d   Preserve last-modified date

-e	Recover from errors and continue
-r[*n*]	*Recur n* levels only (default, one)
-s*name*	Suppress copy of *name*, which is relative to *dir1*
-t	Test and report errors — do not change anything
-u	Update regular files if more recent
-v	Verbose file by file account

cpio — Archiving/backup utility

cpp — C preprocessor
/lib/cpp [*option...*] [*file...*]

Options:

-C	Do not suppress comments
-D*VARIABLE***[=***value***]**	
	Define *VARIABLE*; if *value* is not defined, the default value is one
-E	Strip file line-number information
-I *dir*	Search *dir* for header files
-o *file*	Write output into *file*
-P	Remove file and line-number information
-Q	Suppress all messages
-U*VARIABLE*	Undefine *VARIABLE*
-V	Print verbose information
-VCPLUS	Suppress C++-style on-line comments

crontab — Copy a command file into the crontab directory
/usr/bin/crontab [-l] [-r] [-f *filename***] [-m[ed]] [-u***user***]**

Option:

-f *filename*	Replace a user's command file.
-l	List your command file.
-m[ed]	Enable/disable sending mail upon failure of a command within a command file.
-r	Remove user's command file.
-u*user*	Specify that the file being copied is to be applied to *user*. Only the superuser **root** can execute this option.

crypt — Encrypt/decrypt text
crypt [*password***]**
Password is ten characters or fewer. The same password encrypts and decrypts.

ctags — Generate tags and refs files for vi editor
ctags [**-r**] *files...*

cu — -compatible communications utility
cu [options] [*system***] [***phone***] [dir]**

-a *port*	Use named port
-c *phone*	Phone number to call
-d	Set maximum debugging level
-e	Set even parity
-h	Echo locally
-I *file*	Set configuration file to use
-l *line*	Use named device (e.g. tty0)
-n	Prompt for telephone number
-o	Set odd parity
-p *port*	Use named port
-s *speed, -#*	Use given speed
-t	Map carriage return to carriage return/linefeed
-x *activity*	Log a given *activity*, for debugging purposes
-z *system*	System to call

cut — Select portions of each line of its input
cut -c*list* [*file ...*]
cut -f*list* [**-s**] [**-d** *char*] [*file ...*]

Options:

-c*list*	*list* specifies character positions
-d*char*	Use *char* as field delimiter
-f*list*	*list* specifies fields
-s	Suppress every line that does not contain a field delimiter

cvmail — Convert mail from 3.X format to SV format
cvmail [-m *filename*] [*filename*]

date — Print/set the date and time
date [-s] [-u] [[*yymmdd*]*hhmm*[*.ss*]]
Options:

-s	Suppress daylight savings time conversion
-u	Print (and enter) date in Greenwich Mean Time

db — Assembler-level symbolic debugger
db [-a *symfile*] [**-cdefort**] [[*mapfile*] *program*]
Options:

-a *symfile.*	Use *symfile* as the symbol file, instead of reading the executable's symbol table.
-c	Map *program* as a core file
-d	Map *program* as a system dump; *mapfile* defaults to **/coherent**
-e	Next argument is object file and rest of command line is passed to the child process
-f	Map *program* as binary data
-k	Map program as a kernel process; *mapfile* defaults to **/coherent**
-o	*program* is an object file
-p *prompt*	Change the command prompt from **db:** to *prompt*
-r	Access all files read-only
-s	Do not load symbol table.
-t	Perform input and output via **/dev/tty** rather than stdin and stdout

By default, *program* is assumed to be an object file. *mapfile* defaults to **l.out** or **a.out**, and *program* defaults to **core**.

dc — Desk calculator
dc [*file*]
Arbitrary precision desk calculator with registers, using reverse-Polish notation. Reads input from *file* if given, then from stdin.

dcheck — Check directory consistency
dcheck [-s] [-i *inumber...*] *filesystem* ...
Options:

-s	Cause **dcheck** to correct link counts automatically
-i	Print information about each given i-number

dd — Convert the contents of a file
dd [*option=value*] ...
Options:

bs=*n*	Set I/O buffer size to *n*
cbs=*n*	Set conversion buffer size to *n*
conv=*list*	Comma-separated list of conversions:
ascii	Convert EBCDIC to ASCII
ebcdic	Convert ASCII to standard EBCDIC
ibm	Convert ASCII to IBM print codes
lcase	Map all letters to lower case
noerror	Continue if error occurs
swab	Swap byte pairs
sync	Pad input to *ibs*
ucase	Map all letters to upper case
count=*n*	Number of buffers to copy from input
files=*n*	Number of files to copy (useful with tape)
ibs=*n*	Input buffer size
if=*file*	Set input file to *file*
obs=*n*	Set output block size to *n*
of=*file*	Set output file to *file*
seek=*n*	Set seek position of output to *n*
skip=*n*	Skip *n* input blocks

deroff — Remove text formatting control information
deroff [-w] [-x] [*file* ...]
Options:

-w	Divide the output into words, one per line

-x	Extra knowledge of macro packages

detab — Replace tab characters with spaces
detab [*tabsize***]**

Option:

tabsize	Set *tabsize* (2-256, inclusive)

df — Measure free space on disk
df [-fiv] [-t*filesys***]** (default format)

Options:

-f	Suppress i-node information
-i	Give percentage of i-nodes used
-v	Give percentage of blocks used

If the command line does not name a *filesys*, **df** prints information about the file system that you are in.

diff — Compare two files
diff [-bdefh] [-c *symbol***]** *file1 file2*

Options:

-b	Ignore trailing blanks; all strings of blanks are equal
-c *sym*	Make **cpp** input conditionalized on *sym*
-d	Use **-h** algorithm for large (>25,000 character) files
-e	Make **ed** script
-f	Make fake (non-usable) **ed** script
-h	Half-hearted algorithm (works on long files)
-s	Make **sed** script

If either *file1* or *file2* is '-', stdin is used. If one *file* is a directory, the other *file* under that directory is used.

diff3 — Summarize differences among three files
diff3 [-ex3] *file1 file2 file3*

Options:

-e	Make **ed** script to change *file2* and *file* to *file1* (changes marked with **====** or **====3**)
-x	Above script with changes marked **====** (all different)
-3	Above script with changes marked **====3** (*file3* different)

dirname — Extract a directory name
dirname *string*

dirs — Print the contents of the directory stack
dirs
sh only.

disable — Disable a port
/etc/disable *port...*

dos — Manipulate files on MS-DOS file systems
dos [-]dFflrtx[*flags***] [***device***] [***file ...***]**

Commands:

d	Delete specified files
F	Build file system (format)
f	Force removal of readonly files on the DOS side
l[*label***]**	Label disk
r	Replace *files* (default, all files in '.')
t	List contents (default, all files)
x	Extract specified *files* (default, all files)

Flags:

a	ASCII data extract/replace (default, binary data)
c	Read only; do not write changes to file system
k	Keep *mtime* on extract/replace (default, now)
n	Newest files first in list (default, alphabetized)
p	Piped extract/replace
s*dir*	Suppress subdirectory *dir*
v	Verbose
[1-9]	Specify logical drive on extended partition

The default device is **/dev/dos**.

doscat — Concatenate a file on an file system
doscat *device***:[/***directory***/]***file*

doscp — Copy files to/from an file system
doscp [-abkmrv] *src dest*

Options

a	ASCII. When copying from to , convert the carriage-return/newline combination to newline characters; when copying from to , do the opposite.
b	Binary. Do not convert newline conversion.
k	Keep the time stamp on copied files.
m	Move. Same as **a**, described above.
r	Same as **b**, described above.
v	Verbose. Describe each action as it is executed.

doscpdir — Copy a directory to/from an file system
doscpdir [-akmv] *src dest*

Options

a	ASCII. When copying from to , convert the carriage-return/newline combination to newline characters; when copying from to , do the opposite.
k	Keep the time stamp on copied files.
m	Move. Same as **a**, described above.
v	Verbose. Describe each action as it is executed.

dosdel — Delete a file from an file system
dosdel [-fv] *device***:/***dir***/***file*

Option

f	Force removal of **readonly** files
v	Verbose. Describe each action as it is executed.

dosdir — List contents of an directory
dosdir [-nv] *device***:[***dir***/][***file***]**

Options

n	List files in order of creation (newest file last) rather than in alphabetical order.
v	Verbose. Describe each action as it is executed.

dosformat — Build an MS-DOS file system
dosformat [-v] *device***:**

Option

v	Verbose. Describe each action as it is executed.

doslabel — Label an floppy disk
doslabel [-v] *device***:** *label*

Option

v	Verbose. Describe each action as it is executed.

dosls — List files on an file system
dosls [-v] *device***:[/***directory***/][***file***]**

Option:

-v	Print output in long format, analogous to **ls -l**.

dosmkdir — Create a directory in an file system
dosmkdir *device***:***directory*

dosrm — Remove a file from an file system
dosrm *device***:[/***directory***/]***file*

dosrmdir — Remove a directory from an file system
dosrmdir *device***:***directory*

dpac — De-fragment a COHERENT file system
dpac [-q] *raw_device*

Option:

-q	Run quietly.

du — Summarize disk usage
du [-a] [-s] [*directory* ...]**

Options:

-a	Print an entry for each file
-s	Print only a summary

dump — File-system backup utility
dump [*options***] [***argument* ...]**

Options:

0-9	Set dump level (default, 9)
b	Next argument is blocking factor (default, 20)
d	Next argument is density in bpi (default, 1600)
f	Next argument is output device name
s	Next argument is tape length in feet (default, 2300)
S	Next argument is floppy disk size in blocks
u	Update **/etc/ddate**
v	Verbose (display date and tape length)

dumpdate — Print dump dates
dumpdate [*filesystem* ...]**

dumpdir — Print the directory of a dump
dumpdir [af [*argument* ...]]**

Options:

a	List normally suppressed '.' and '..' names
f	Next argument is dump device name (default, **/dev/dump**)

echo — Repeat/expand an argument
echo [-n] [*argument* ...]**

Option:

-n	Do not print terminal newline

Copies all command arguments to the standard output, with the following special-character sequences being replaced with the equivalent ASCII character:

\b	Backspace
\c	Print line without a newline (like **-n** option)
\f	Formfeed
\n	Newline
\r	Carriage return
\t	Tab
\v	Vertical tab
****	Backslash
nnn	*nnn* is octal value of desired character

ed — Interactive line editor
ed [-] [+cmopsv] [*file***]**

Options:

-	Suppress character counts on **r, w, e** commands
-x	Encrypt *file*
+c	Print character counts on **r, w, e**
+m	Allow multiple commands per line
+o	Print line counts instead of character counts
+p	Prompt for each command with '*'
+s	Lower case matches upper in patterns
+v	Verbose error messages

egrep — Extended pattern search
egrep [-Abcefhily] [*pattern***] [***file* …]**

Options:

-A	Build *error list* for interactive editing using , like **-A** option to the **cc** command
-b	Each output line has block number of match
-c	Print only a count of the matching lines
-e	Next argument is *pattern*
-f	Next argument is file with one pattern per line
-h	Suppress printing of file names on matched lines
-i	Ignore case when matching alphabetic characters in *pattern*

-l	Print only names of files containing matches
-n	Print line number of file with each matched line output
-s	Suppress output, just return status
-v	Negate the sense of match
-y	Lower-case letters in *pattern* match upper- and lower-case

The *pattern* is a pattern roughly like that found in **ed**. If no *file* is specified, stdin is read. **egrep** is like **grep -a**, but is an order of magnitude faster.

elvis — Clone of Berkeley-standard screen editor
elvis [*options* **] [+***cmd* **] [** *file1 ... file27* **]**

Options:

-e	Begin in colon-command mode
-i	Begin in input mode
-r	Recover a previous edit
-R	Read-only mode
-s	Turn off some options, so that **elvis** runs more safely
-t *tag*	Begin editing at *tag*
-m	Use in error-handling mode
-v	Begin in visual-command mode
-w *winsize*	Set option **window** to *winsize*
+*command*	Execute *command* before editing

elvprsv — Preserve the modified version of a file after a crash
elvprsv ["-*why elvis died***"] /tmp/***filename***...**
elvprsv -R /tmp/*filename***...**

elvrec — Recover the modified version of a file after a crash
elvrec [*preservedfile* **[***newfile***]]**

emacs — screen editor
emacs [-e *errorfile***] [-f** *bindfile***] [***textfile ...***]**

enable — Enable a port
/etc/enable *port...*

env — Execute a command in an environment
env [-] [*VARIABLE***=***value ...***] [***command args***]**

epson — Prepare files for Epson printer
epson [-cdfnrw8] [-b *head* **] [-i** *n* **] [-o** *file* **] [-s** *n* **] [** *file ...* **]**

Options:

b *head*	Print wide banner *head* at top of first page
c	Compressed printing
d	Print boldface with double strike, not emphasize mode
f	Suppress formfeed after each *file*
i*n*	Indent output 'n' spaces
n	Suppress italics escape sequence in initialization string
o *file*	Write output into *file*, instead of sending it to device **/dev/lp**
r	Use only Roman character set (no italics)
s*n*	Vertical spacing *n* (default, 1)
w	Double-width printing
8	Eight lines per inch (default, six)

eval — Evaluate arguments
eval [*token ...*]
The shell executes **eval** directly.

ex — Berkeley-style line editor
ex [*options* **] [+***cmd* **] [** *file1 ... file27* **]**

Options:

-r	Recover a previous edit
-R	Read-only mode
-t *tag*	Begin editing at *tag*
-m	Use in error-handling mode
+*command*	Execute *command* before editing

exec — Execute command directly
exec [*command*]
The shell executes *command* by one of the **exec()** functions rather than
through the system call **fork()**. This normally terminates the current shell.
Current shell I/O may be redirected by **exec** with no *command*.

exit — Exit from a shell
exit [status**]**
The previous status is retained if none is specified. **exit** sets the status
but does not terminate an interactive shell. The shell executes **exit**
directly.

export — Add a shell variable to the environment
export [name ...**]**
export [name=value**]**

expr — Compute a command-line expression
expr *argument ...*

Options:
n	Any integer with optional sign
string	Used with comparisons and **len** operator
+	Arithmetic operators (one of '+', '-', '*', '/', '%')
!	Unary not
-	Unary minus
==	Relational operators (one of '>', '<', '>=', '<=', '==', '!=')
&	Logical AND of previous and next expression
|	Logical OR of previous and next expression
len	Length of string given by next argument
e1:e2	Set to number of characters matching regular expression *e2* in string *e1*; if *e2* contains any '\(...\)' sequences, result is concatenation of matched parts
(*e* **)**	Parentheses for grouping
{	
	Evaluate *e2* if *e1* is true, *e3* otherwise; *e3* defaults to 0 if missing

factor — Factor a number
factor [*number ...* **]**

false — Unconditional failure
false

fc — Edit and re-execute one or more previous commands
fc [-e *editor*] **[-ln] [***first* **[***last***]]**

Options:
-l	Print commands on stdout
-e *editor*	Use *editor* instead of the editor named in the evironmental variable **FCEDIT**
-n	Suppress default command numbers

ksh only.

fdformat — Low-level format a floppy disk
/etc/fdformat [*option ...* **]** *special*

Options:
-a	Print information on stdout during format
-i *n*	Interleave factor *n* (0-7; default, 6)
-o *n*	Skew factor *n* for sector numbering (default, 0)
-v	Verify
-w *file*	Copy *file* to formatted floppy disk track by track

fdisk — Hard-disk partitioning utility
/etc/fdisk [-r] [-c] [-b *mboot***]** *xdev ...*

Options:
-r	Read-only access
-b	Add master boot code from file *mboot*
-c	Specify disk geometry for non-standard drives
-V	Display the version number of **fdisk**

A hard disk can be split into a maximum of four partitions (logical devices).

file — Guess a file's type
file *file ...*

find — Search for files satisfying a pattern
find *directory ...* **[***expression ...***]**
Expression:
   -atime *n*   File has been accessed in *n* days
   -ctime *n*   File's i-node has been changed in *n* days
   -exec *cmd*   Command *cmd* is successful
   -group *gn*   File belongs to group *gn*
   -inum *n*   File has i-node *n*
   -links *n*   File has *n* links to it
   -mtime *n*   File has been modified within *n* days
   -name *pattern* File name matches *pattern* (shell conventions)
   -newer*file*   File has been modified since *file*
   -nop   Always true; does nothing
   -ok *cmd*   Like **-exec**, except it asks
   -perm *octal*   File permissions are *octal*
   -print   Always true; prints current path name
   -size *n*   File is *n* blocks long
   -type *c*   File matches type (*c* may be [bcdfmp])
   -user *uname* *uname* owns file
   exp -a *exp exp* Both expressions are true
   exp -o *exp exp* One of the expressions is true
   ! *exp*   Expression is false
   (*exp***)**   Parentheses for grouping
If no expression is specified, **-print** is assumed.

findmouse — Examine a port to see if a mouse is plugged into it
/usr/local/bin/findmouse *port*

fmap — Measure fragmentation of the free list
fmap *device*

fmt — Adjust the length of lines in a file of text
fmt [-*width***] [***textfile ... textfile***]**

fnkey — Set/print function keys for the console
fnkey [*n* **[** *string* **]]**
Sets function key *n* to send *string*; if no *string*, set it to send nothing. If no
arguments, **fnkey** prints the function keys.

for — Execute commands for tokens in list
for *name* **[in** *token ...***] do** *sequence* **done**
If *in* clause is omitted, list of positional parameters to current script is
assumed. Both **do** and **done** must be first token on line or preceded by ';'.
The shell executes **for** directly.

fortune — Print randomly selected, hopefully humorous, text
/usr/games/fortune [*file* **]**

Option:
   file   Read a fortune from *file*, instead of the default file
                /usr/games/lib/fortunes

from — Generate list of numbers, for use in loop
from *start* **to** *stop* **[by** *incr* **]**
start, *stop*, and *incr* (default, one) are decimal integers with optional '-'.

fsck — Check and repair file systems interactively
/etc/fsck [**-fnqsSy**] [**-t** *tempfile*] [*filesystem ...*]
Options:
   -f   Fast check; check only if a block is claimed by more
                than one i-node, by an i-node and the free list, or more
                than once in the free list
   -n   Default reply of no to all queries
   -q   Quiet option; syppress file name warning messages
   -s   Force reconstruction of the free list for all unmounted
                file systems.
   -S   Same as **-s**, but works on mounted file systems as well

-t	Use *tempfile* for temporary storage
-y	Default reply of yes to all queries

ftbad — Manipulate bad-block list on a floppy-tape cartridge
ftbad [-rw] [*device***]**

Options:

-r	Read the list of bad blocks and display it on the standard-output deviec
-w	Read a list of bad blocks from the standard input and write it onto the floppy-tape cartridge

fwtable — Build font-width table
fwtable [-ptv] [*infile* **[** *outfile* **]]**

Options:

-p	*infile* is PostScript AFM file, not PCL bitmap font
-t	*infile* is a Hewlett-Packard **.tfm** file, which describes a font that is built into the Hewlett-Packard LaserJet III
-v	Write a brief font description to **stderr**

gawk — Pattern-scanning and -processing language
gawk [*POSIX or GNU style options* **] -f** *program-file* **[--]** *file ...*
gawk [*POSIX or GNU style options* **] [--]** *program-text file ...*

getmap — De-archive Usenet map articles
/usr/lib/mail/getmap [-b *batchfile***] [-m** *mapdir***] [-n** *newsgroup***] [-u**
*username***] [-w** *workdir***]**

Options

-b *batch*	De-archive *batch* (default, **/usr/spool/uumaps/work/batch**)
-m *mapdir*	Copy articles into *mapdir* (default, **/usr/spool/uumaps**)
-n *newsgroups*	Read articles from *newsgroup*
-u *user*	Mail errors to *user* (default, **postmaster**)
-w *workdir*	Keep logs and batch files in *workdir* (default, **/usr/spool/uumaps/work**)

getopts — Parse command-line options
getopts *optstring name [opt]*

gnucpio — Archiving/backup utility
Copy-in mode: **cpio** {**-o** | **--create**} **[-0acvABLV] [-C** *bytes***] [-H** *format***] [-M**
*message***] [-O [[***user***@]***host***:]***archive***] [-F [[***user***@]***host***:]***archive***]** [--
file=[[*user***@**]*host***:**]*archive***] [--format=***format***] [--message=***message***] [--null]**
[--reset-access-time] [--verbose] [--dot] [--append] [--block-size=*blocks***] [--**
dereference] [--io-size=*bytes***] [--version] <** *name-list* **[>** *archive***]**
Copy-out mode: **cpio** {**-i** | **--extract**} **[-bcdfmnrtsuvBSV] [-C** *bytes***] [-E** *file***] [-**
H *format***] [-M** *message***] [-R [***user***][:.][***group***]] [-I [[***user***@]***host***:]***archive***] [-F**
[[*user***@]***host***:]***archive***] [--file=[[***user***@]***host***:]***archive***] [--make-directories] [--**
nonmatching] [--preserve-modification-time][--numeric-uid-gid][--
rename] [--list] [--swap-bytes] [--swap] [--dot] [--unconditional] [--verbose]
[--block-size=*blocks***] [--swap-halfwords] [--io-size=***bytes***] [--pattern-**
file=*file***] [--format=***format***] [--owner=[***user***][:.][***group***]] [--no-preserve-**
owner] [--message=*message***] [--version] [***pattern...***] [<** *archive***]**
Copy-through mode: **cpio** {**-p** | **--pass-through**} **[-0adlmuvLV] [-R**
[*user***][:.][***group***]] [--null] [--reset-access-time] [--make-directories] [--link]**
[--preserve-modification-time][--unconditional] [--verbose] [--dot] [--
dereference] [--owner=[*user***][:.][***group***]] [--no-preserve-owner] [--version]**
destination-directory **<** *name-list*

Options:

-a	Reset access time of input files after copying
-B	Change size of a block
-c	Write header information in ASCII rather than binary
-d	Create directories as needed
-f*pattern*	Copy all files except those matching *pattern*
-i	Read the standard input
-l	Link files rather than copying them
-m	Retain previous modification times
-o*pattern*	Copy all files matching *pattern*
-p	Read stdin for files names to copy to destination
-r	Interactively rename files
-t	Print table of contents of an existing archive
-u	Copy files unconditionally

    -v   Verbose output

grep — Pattern search

Options:

-a	Extra metacharacters supported ('(...)', '|', '+', and '?')
-b	Each output line has block number of match
-c	Print only count of matching lines
-e	Next argument is pattern
-f	Next argument is file containing one pattern per line
-h	Suppress printing of file names on matched lines
-i	Ignore case when matching letters in *pattern*
-l	Print only names of files containing matches
-n	Print line number of file with each matched line output
-s	Suppress output, just return status
-v	Negate sense of match
-x	Exact match (don't expand metacharacters)
-y	Lower-case letters in *pattern* match only upper-case

The *pattern* is a regular expression roughly like that found in **ed**. If no *file* is specified, stdin is read.

gtar — Archiving/backup utility

gtar *options*

Options:

-A	
--catenate	
--concatenate	Append files to an archive
--atime-preserve	
	Don't change access times on dumped files
-b	
--block-size *N*	Block size of *N*×512 bytes (default, *N*=20)
-B	
--read-full-blocks	
	Reblock as we read (for reading 4.2BSD pipes)
--block-compress	
	Block the output of compression program for tapes
-c	
--create	Create a new archive
-C	
--directory *dir*	Change to directory *dir*
--checkpoint	Print directory names while reading the archive
-d	
--diff	
--compare	Find differences between archive and file system
--delete	Delete from the archive
--exclude *file*	Exclude *file*; *file* can be a regular expression
-f *file*	
--file *file*	Use archive file or device *file* (default, the standard input)
--force-local	Archive file is local even if has a colon
-F *script*	
--info-script *script*	
--new-volume-script *script*	
	Run *script* at end of each tape (implies **-M** option)
-G	
--incremental	Create/list/extract old GNU-format incremental backup
-g *file*	
--listed-incremental *file*	
	Create/list/extract new GNU-format incremental backup
-i	
--ignore-zeros	Ignore blocks of zeros in archive (normally mean EOF)
--ignore-failed-read	
	Don't exit with non-zero status when an unreadable file is encountered
-k	
--keep-old-files	Keep existing files; don't overwrite them from archive
-K *file*	
--starting-file *file*	
	Begin at *file* in the archive
-l	
--one-file-system	
	Stay in local file system when creating an archive

-L *N*
--tape-length *N*Change tapes after writing *N*×1024 bytes
-m
--modification-time
                    Don't extract file modified time
-M
--multi-volume Create/list/extract multi-volume archive
-N *date*
--after-date *date*
--newer *date*   Store only the files newer than *date*
--null   **-T** reads null-terminated names, disable **-C**
-o
--old-archive
--portability   Write a V7 format archive, rather than ANSI format
-O
--to-stdout   Extract files to standard output
-p
--same-permissions
--preserve-permissions
                    Extract all protection information
-P
--absolute-paths
                    Don't strip leading '/'s from file names
--preserve   Combine options **-p** and **-s**
-r
--append   Append files to the end of an archive
-R
--record-number
                    Show record number within archive with each message
--remove-files   Remove files after adding them to the archive
-s
--same-order
--preserve-order
                    List of names to extract is sorted to match archive
--same-owner   Create extracted files with the same ownership
-S
--sparse   Handle sparse files efficiently
-t
--list   List the contents of an archive
-T *file*
--files-from *file* Get names to extract or create from *file*
--totals   Print total bytes written with **-c**
-u
--update   Only append files that are newer than copy in archive
--use-compress-program *program*
                    Filter the archive through *program* (which must accept **-d**)
-v
--verbose   Verbosely list files processed
-V *name*
--label *name*   Create archive with volume *name*
--version   Print **gtar** version number
-w
--interactive
--confirmation Ask for confirmation for every action
-W
--verify   Attempt to verify the archive after writing it
-x
--extract
--get   Extract files from an archive
-X *file*
--exclude-from *file*
                    Exclude all files listed in *file*
-z
--gzip
--ungzip   Filter the archive through **gzip**
-Z
--compress
--uncompress Filter the archive through **compress**

guess — Extraordinarily amusing guessing game
/usr/games/guess

gunzip — GNU utility to uncompress files
gunzip [-cfhLrtvV] [*file ...* **]**

Options:
-c	Write output to the standard output
-f	Force: decompress a *file* even if its has multiple links or the corresponding file already exists
-h	Help: display a screenful of information about how to use this command
-L	Display licensing information
-r	Recurse: if a *file* is a directory, attempt to uncompress all files within it
-t	Test the integrity of a compressed *file*
-v	Give a verbose description of each action
-V	Print the current version of this command

gzip — GNU utility to compress files
gzip [-cdfhLrtvV19] [*file ...* **]**

Options:
-c	Write output to the standard output
-d	Decompress each *file*
-f	Force: compress a *file* even if its has multiple links or the corresponding file already exists
-h	Help: display a screenful of information about how to use this command
-L	Display licensing information
-q	Suppress all warning messages.
-r	Recurse: if a *file* is a directory, attempt to compress all files within it
-t	Test the integrity of a compressed *file*
-v	Give a verbose description of each action
-V	Print the current version of this command
-[1-9]	Regular speed of compression, from **1** to **9**, where **1** performs the fastest compression but most superficial compression, and **9** the slowest but most thorough compression

hash — Add a command to the shell's hash table
hash [-r] [*command ...* **]**

Option:
-r	Remove *command* from hash table

ksh only.

head — Print the beginning of a file
head [+*n***[bcl]] [***file***]**
head [-*n***[bcl]] [***file***]**

Options:
+	Count from beginning of file
-	Count from end of file
b	Count in blocks
c	Count in characters
l	Count in lines

help — Print concise description of command
help [-d*c***] [-f***file***] [-i***file***] [-r] [***command***]...**

Options
-d*c*	Use character *c* as the delimiter between helpfile entries.
-f*file*	Read *file* as the helpfile, instead of the default **/usr/lib/helpfile**.
-i*file*	Read *file* as the helpfile's index, instead of the default **/usr/lib/helpindex**.
-r	Rebuild the helpfile's index.

If *command* is omitted, print information about **$LASTERROR**.

hmon — Monitor the COHERENT System
hmon

hp — Prepare files for Hewlett-Packard LaserJet printer
hp [-acflr] [-i*marg* **] [-t***top* **] [-p***lines* **] [** *file ...* **]**
Options:
    -a   Substitute ' ' for '
    -c   Toggle cartridge in place switch
    -f   Print pages in forward order (default)
    -l   Landscape mode
    -i*marg*   Indent to *marg*
    -p*lines*   Page length is *lines*
    -r   Print pages in reverse order (for LaserJet I).
    -t*marg*   Set top margin to *marg*

hpr — Spool a job for printing on the laser printer
hpr [-Bcemnr] [-b *banner***] [-f** *fontnum***] [***file ...***]**
Options:
    -B   Suppress banner page and extra page at termination.
                    Must be used with a PostScript printer.
    -b   Next argument is the text of the banner
    -c   Make a copy of each *file* in spool area
    -e   Erase all fonts from printer's memory
    -f *fontnum file1 ... fileN*
                    Load into printer memory the HP soft fonts in *file1*
                    through *fileN*; set font identifiers beginning with *fontnum*
    -m   Send a message when listing is complete
    -n   No message (default)
    -r   Remove files when they have been spooled

hpskip — Abort/restart current job on Hewlett-Packard LaserJet
hpskip [-r]
Option:
    -r   Restart the current job
With no argument, terminate the current job.

icheck — i-node consistency check
icheck [-s] [-b *N* **...] [-v]** *filesystem ...*
Options:
    -b   The following numeric arguments give block numbers;
                    all references to these blocks are printed, with type
    -s   Repair file system (requires write access)
    -v   Print summary of information about file system

id — Print user and group IDs and names
id

idbld — Reconfigure the COHERENT kernel
/etc/conf/bin/idbld [-o *kernelname* **]**

ideinfo — Display information of an IDE hard-disk drive
ideinfo [-c] /dev/at*??*

idenable — Enable or disable a device driver
/etc/conf/bin/idenable [-f *file***] [-de]** *driver*

idmkcoh — Build a new kernel
idmkcoh [-o *kernelfile* **]**

idtune — Set a tunable system value
/etc/conf/bin/idtune [-fm] *switch value*

if — Execute a command conditionally
if *sequence1* **then** *sequence2* **[elif** *sequence3* **then** *sequence4***] ... [else** *sequence5***] fi**
Each **then**, **elif**, **else**, and **fi** must occur unquoted at the start of a line or
preceded by ';'. The shell executes **if** directly.

infocmp — De-compile a terminfo file
infocmp [*file ...* **]**

install — Install a software update onto COHERENT
/etc/install [-c] *id device ndisks*

Option:
    -c   Uncompress compressed files.

Arguments:
    id   String that identifies the release; e.g., **coh.301** identifies
                     release version 3.0.1.
    device   The physical device from which the installation takes
                     place; e.g., **/dev/fha0** identifies floppy-disk drive 0
                     (drive A), where it is a high-density, 5.25-inch disk.
    ndisks   Number of floppy disks in the release.

ipcrm — Remove an interprocess-communication memory item
ipcrm [-mqs *id***] [-MQS** *key***]**

Options:
    m *id*   Remove the shared memory segment with an identifier
                 of *id*.
    q *id*   Remove the message queue with an identifier of *id*.
    s *id*   Remove the semaphore set with an identifier of *id*.
    M *key*   Remove the shared memory segment with a key of *key*.
    Q *key*   Remove the message queue with a key of *key*.
    S *key*   Remove the semaphore set with a key of *key*.

ipcs — Display a snapshot of interprocess communications
ipcs [-abcmopst] [-N *kernel***]**

Options:
Use all print options Display the biggest size that the kernel allows for
each kind of IPC object Display login name and group name of the user
who created each IPC object Display information about shared-memory
segments Read kernel-specific information from *kernel* instead of from the
default kernel **/autoboot** Display "outstanding usage" Display process
identifiers Display information about message queues Display information
about sets of semaphores Display the times that each IPC was last used
and modified

jobs — Print information about jobs
jobs
ksh only.

join — Join two data bases
join [-a [*n***]] [-e** *string* **] [-j[***n***]** *key***f] [-o** *n.m ...***] [-t***c***]** *file1 file2*

Options:
    -a[*n***]**   Print unpaired records from file *n*
    -e *s*   Replace empty fields on output with string *s*
    -j[*n***]** *key* Use *key* of file *n* for comparison
    -o [*n.***]***m ..* Subsequent arguments list fields to output; each has file
                n and field number *m*
    -t*c*   Field separator is character *c*

If either *file1* or *file2* is '-', stdin is used. The optional file *n* may be either 1
or 2; if omitted, both 1 and 2 are assumed.

kill — Signal a process
kill [- *signal* **]** *pid ...*

ksh — The Korn shell
ksh *token ...*

l — List directory's contents in long format
l [*file ...***]**

lc — List directory's contents in columnar format
lc [-1abcdfp] [*directory ...***]**

Options:
    -1 (one, not el) List files one per line instead of in columns
    -a   List all files in directory (including '.' and '..')
    -b   List block-special files only

-c	List character-special files only
-d	List directories only
-f	List regular files only
-p	List pipe files only

Options can be combined. If no *directory* is specified, the current directory is used.

lcasep — Convert text to lower case
lcasep [-f *inputfile*] **[-o** *outputfile*]

ld — Link relocatable object modules
ld *[option …] file …*

Options:

-e *ent*	Set entry point to symbol or octal number
-f	(Force) Force link even if there are errors
-i	Bind output sepid
-G	Suppress the common/global warning
-K	Rebuilt a new kernel.
-L*directory*	Search *directory* for libraries and objects before searching the directories named in the environmental variable **LIBPATH**
-l*lib*	Use standard library *lib*
-o *file*	Write output into *file* (default, *l.out*)
-q	(Quiet) Suppress all warning messages
-Q	Quiet: Suppress all error messages
-r	Retain relocation information
-s	Discard symbol table
-u *sym*	Undefine *sym* (force library search)
-x	Discard all local symbols
-X	Discard C internal local symbols

let — Evaluate an expression
let [expression**]**
ksh only.

lex — Lexical analyzer generator
lex [-t][-v][file**]**
cc lex.yy.c -ll

Options:

-t	Write to standard output instead of **lex.yy.c**
-v	Give statistics about generated tables

lf — List directory's contents in columnar format
lf [file ...**]**

lines — Highly amusing board game
/usr/games/lines

lmail — Deliver mail on your local system

ln — Create a link to a file
ln [-f] *oldfile newfile*
ln [-f] *oldfile ... directory*

Option:

-f	Force link even if *newfile* exists

login — Log in a user
login [-p] [*login_id* **[***environ_var***[=***value***] …]]**

Options:

-p	Preserve the current environment.
login_id	The user's login identifier.
environ_var	An environment variable to set upon logging in.

look — Find matching lines in a sorted file
look [-df] *string* [*file*]

Options:

-d	Dictionary ordering
-f	Fold cases for comparison

If no *file*, **look** uses **/usr/dict/words** with **-df** option.

lp — Spool a file for printing
lp [-d*printer***] [-t** *title***] [-n***copies***] [-R** *page* [*page*]] **[-Smws]** *file ...*
Options:
   -R *request*   Print a job beginning from the first *page* and continuing
                           either to the second *page* or to the end of the document
                           (if no second *page* is specified)
   -S   Shut down the spooler daemon **lpsched**
   -d*printer*   Print *file1* through *fileN* on *printer*
   -m   Send mail to the user when the spooled job is printed
   -n*copies*   Print *copies* copies of *file1* through *fileN*
   -s   Silent — do not acknowlege submissions
   -t *title*   Give this job *title*
   -w   Write to the user when the request is printed

lpadmin — Administer the lp print-spooler system
lpadmin [-d*printer***]**
lpadmin [-p*printer***] [-v***device***] [-m***backend***]**
lpadmin [-x*printer***]**

Options:
   -d*printer*   Make *printer* the default system printer
   -m*script*   When used with option **-p**, use *script* to preprocess text
                           to be printed on *printer*
   -p*printer*   Use *printer* with options **-m** or **-v**
   -v*device*   Associate *device* (a serial or parallel port) with the
                           printer with **-p**, above
   -x*printer*   Delete *printer* from the system

lpr — Spool a job for printing on the line printer
lpr [-cmnr] [-b *banner***] [***file ...***]**
Options:
   -B   Suppress printing of a banner. This option **must** be
                           used with PostScript printers.
   -b   Next argument is the banner
   -c   Copy each *file* in spool area
   -m   Send a message when listing is complete
   -n   No message (default)
   -r   Remove files when they have been spooled

lpsched — Print jobs spooled with command lp; turn on printer daemon
lpsched

lpshut — Turn off the printer daemon despooler
lpshut [-d]
Option **-d** tells **lpshut** to finish currently printing jobs

lpskip — Abort/restart current job on line printer
lpskip [-r]
Option:
   -r   Restart current job
With no argument, terminate current job.

lpstat — Give status of printer or job
lpstat [-p*printer***] [-drqstv]**
Options
   -p *printer*   Give the status of *printer*
   -d   Name the system default destination printer
   -r   Give the status of the daemon **lpsched**
   -q   Give a detailed report of jobs in the queue
   -s   Summarize status of each request and status of each
                           printer
   -t   Like option **-s**, but in somewhat more detail
   -v   List all available printers and the devices associated
                           with them

lr — List subdirectories' contents in columnar format
lr [*file ...***]**

ls — List directory's contents
ls [-abCcdFfgilmnopqRrstux][*file ...* **]**

Options:

-a	List all files (including '.' and '..')
-b	Print non-graphic characters in octal
-C	Print output in multi-column format, sorted down the columns
-c	Use attribute change instead of modified time for **-l** and **-t**
-d	Treat directories like files
-F	Print '/' after directories and '*' after executables
-f	Treat *file* as a directory even if it is not
-i	Print the i-number as well
-l	Long format: show file type, permissions, size
-m	Output file names separated by commas
-n	Same as **-l**
-p	Print '/' after directory names
-q	Print non-graphic characters as '?'
-R	Recursively display directories
-r	Reverse the order of all sorting
-s	Print the file size in blocks as well
-t	Sort by times, newest first
-u	Use accessed rather than modified time
-x	Print multicolumn output, sorted across the columns

lx — List directory's contents in columnar format
lx [*file ...***]**

m4 — Macro processor
m4 [file …]
If *file* is '-' or omitted, **m4** reads the standard input.

mail — Send or read mail
mail [-mpqrv] [-f *file***] [***user ...***]**

Options:

-f *file*	Print mail from *file* instead of default
-m	Notify each logged-in recipient when mail is sent
-p	Print mail non-interactively
-q	Exit on interrupt, leaving mail unchanged
-r	Print mail in reverse order
-v	Verbose mode: show version and expanded aliases

If *user* is present, send each a mail message read from standard input.
Mail message ends with EOF, a line containing only '.', or a line containing
only '?'; the last moves the message into editor for further editing
processing before transmission.

Commands:

d	Delete current message; print the next
m [*user ...*]	Mail current message to each *user*
p	Print this message again
q	Quit and update mailbox
r	Reverse direction of scan through mailbox
s [*file ...*]	Save current message with header in each *file*
t [*user ...*]	Send message from stdin to each *user*
w [*file ...*]	Write current message without header in each *file*
x	Exit without updating mailbox
<newline>	Print the next message
-	Print the previous message
EOF	Quit and update mailbox; same as **q** command
?	Print a command summary
!*command*	Pass *command* to the shell for execution

mailq — Display information about spooled mail
mailq [-v]

make — Program-building discipline
make [*option …***] [***argument …***] [***target …***]**

Options:

-d	Debug mode
-e	Macro definitions in environment override those in **makefile**

-**f** *file*	Instructions are in *file* (default, **[mM]akefile**)
-**i**	Ignore command error returns
-**k**	Continue to update other targets that do not depend upon the current target if a non-ignored error occurs while executing the commands to bring a target up to date
-**n**	Test: do all but execute commands
-**p**	Print macro definitions and target descriptions
-**q**	Only return exit status (zero if files up to date)
-**r**	Ignore built-in rules
-**S**	Terminate **make** if an error occurs while executing the commands to bring a target up to date
-**s**	Do not print command lines when executed
-**t**	Update times of files without regenerating

makeboot — Create a bootable floppy disk
makeboot

man — Display Lexicon entries
man [-dw] [*page ...*]

Options:

-**d**	Dump a listing of all available manual pages
-**w**	Print only file name where document resides
-**V**	Print the version of **man** that you are running.

me — screen editor
me [-e *errorfile*] **[-f** *bindfile*] **[***textfile ...***]**

Options:

-**e** *errorfile*	Error-handling mode; read error messages from *errorfile*
-**f** *bindfile*	Read keyboard bindings from *bindfile*

mesg — Permit/deny messages from other users
mesg [ny]

Options:

n	Disallow messages
y	Allow messages

With no arguments, **mesg** prints the state.

mkdbm — Build a data base for smail
/usr/lib/mail/mkdbm [-d] [-f] [-n] [-o *output-file*] **[-v] [-y] [***file ...***]**

Options:

-**d**	Suppress writing the extra '@' record
-**f**	Fold the key to lower case before storing it within the data base
-**n**	Suppress writing a NUL character at the end of each line
-**o** *output-file*	Write output into file *output-file*
-**v**	Write statistics to the standard output
-**y**	Create a YP-compatible output file

mkdir — Create a directory
mkdir [-rp] *directory*

Option:

-**r**	Make parent directories recursively as required
-**p**	Same as -**r**; included for reasons of compatibility

mkfnames — Generate data base of user names
/usr/bin/mkfnames [*namefile ...***]**

mkfs — Make a new file system
/etc/mkfs [-b *boot*] **[-d] [-f** *name*] **[-i** *inodes*] **[-m** *arg*] **[-n** *arg*] **[-p** *pack*] *filesystem proto*

Options:

-**b** *boot*	Specifies the file to use as the "bootstrap" for the file system.
-**d**	Preserve file dates and times.
-**f** *name*	Label the file system with the given *name*. *name* must be less than seven characters in length.
-**i** *inodes*	Use *inodes* as the number of i-nodes for the file system.
-**m** *arg*	Number of blocks to skip when incrementing virtual block number

-n *arg*	Size of a "virtual cylinder"
-p *pack*	Set the file system "pack name" to *pack*. *pack* must be less than seven characters in length.

If *proto* is a number, it is the size in blocks of an empty file system; otherwise, it names a prototype description file, as created by the command **badscan**.

mkhpath — Build a pathalias data base from a hosts table
/usr/lib/mail/mkhostpath [-d] [-c *cost***] [-g** *gateway***] [-n** *netname***] [- ∣** *filename* **]**

Options

-c *cost*	Set the cost of accessing the gateway to *cost*
-d	Print a line only if it contains a domain host name
-g *gateway*	Make *gateway* the gateway to the hosts
-n *netname*	Form a network map and name the network *netname*

mkline — Fold an alias file, paths file, or mailing list into one-line records
/usr/lib/mail/mkline [-ltn] [*file ...* **]**

Options:

-l	Generate a list of addresses
-n	Do not extract keys from the input
-t	Separate the key from the data with a single tab character

mklost+found — Make an enlarged lost+found directory
/etc/mklost+found *directory* **[***slots***]**

mknod — Make a special file or named pipe
/etc/mknod [-f] *filename type major minor*
/etc/mknod [-f] *filename* **p**

Option

-f	Forces creation of a new node, even if one of same name already exists

In first form of the command, *type* is 'b' for block special or 'c' for character special; *major* and *minor* are numbers. The second form creates a named pipe with the given *filename*.

mkpath — Create a pathalias output file
**/usr/lib/mail/mkpath [-v] [-V] [-x] [-e] [-n] **
                              [-t *trace***] [***path_config***]**

Options

-e	Stop when encountering a syntax error, or if a command that it invokes exits with a non-zero status
-n	Disable execution of any commands useful with Bourne shell's option **-v**
-t *tracefile*	Copy into *tracefile* all data passed to command **pathalias**
-V	Invoke command **pathalias** with its option **-v**
-v	Verbose mode — invoke with **sh**'s option **-v**
-x	Verbose mode — invoke with **sh**'s option **-x**

mksort — Sort the standard input, allowing arbitrarily long lines
/usr/lib/mail/mksort [-f] [*file ...* **]**

Options:

-f	Fold upper-case text into lower case

moo — Greatly amusing numeric guessing game
/usr/games/moo [*numdigits* **]**

more — Display text one page at a time
more [-cdflsu] [-*window_size* **] [+***line_number* **] [+/***pattern* **] [** *file ...* **] [-]**

Options:

-	Read/display stdin
-c	Paint screen from top down
-d	Prompt user to quit after each screenful of text
-f	Count lines from file rather than screen-display lines
-l	Do not treat **<ctrl-L>** as special
-s	Squeeze consecutive blank lines into one
-u	Display backspaces as control characters
+line_number	
	Begin display at *line_number*

+/pattern   Begin display at first line to contain *pattern*

mount — Mount a file system
/etc/mount [*device directory* **[-ru]]**
Options:
   -r   Mount device read-only
   -u   Update **/etc/mtab** entry but do not mount device
With no arguments, print devices currently mounted. *device* names a
device-special file; *directory* names the directory on which to mount it. File
/bin/mount contains useful abbreviations for invoking **/etc/mount**.

msg — Send a brief message to other users
msg *user*
message

msgs — Read messages intended for all users
msgs [-*q*] [*number*]
Options:
   -q   Query if new message is waiting to be read
   number   Print message titled with given *number*
To submit a message to **msgs**, mail it to user **msgs**.

mv — Rename files or directories
mv [-f] *oldfile* **[***newfile***]**
mv [-f] *file ... directory*
Option:
   -f   Force: remove *newfile* even if unwritable

mvdir — Rename a directory
/etc/mvdir *olddir newdir*

mwcbbs — Download files from the Mark Williams bulletin board
mwcbbs [-cp] [-d*path***]** *directory*
Options:
   -c   Force **uucp** to telephone the Mark Williams bulletin
                board immediately
   -d*path*   Use *path* in place of the default receive path
   -p   Print the **Contents** file

ncheck — Print file names corresponding to i-node
ncheck [-i *number* **...] [-as]** *filesystem ...*
Options:
   -a   Print file names including '.' and '..'
   -i *n...*   Print file names only for listed i-numbers *n...*
   -s   Print only special files and files with setuid mode

newaliases — Build the smail aliases data base from an ASCII source file
/usr/lib/mail/newaliases

newgrp — Change to a new group
newgrp *group*

newusr — Add new user to system
/etc/newusr *login "User Name" parentdir* **[** *shell* **]**

nm — Print a program's symbol table
nm [-**adgnopru**] *file ...*
Options:
   -a   Print all symbols
   -d   Print only definitions
   -g   Print only global symbols
   -n   Sort numerically (default, sort by name)
   -o   Prepend file or member name to each line
   -p   Print in symbol table order (no sort)
   -r   Reverse order of sort
   -u   Print undefined symbols
file may be an object file or an archive.

nohup — Run a command immune to hangups and quits
nohup *command* **[***arguments***]**

nptx — Generate permutations of users' full names
/usr/bin/nptx

nroff — Text-formatting language
nroff **[***option ...***] [***file ...***]**
Options:
   -d   Debug: print each request before executing
   -f *name*   Write temporary file in file *name*
   -i   Read stdin after each *file* has been read
   -k   Keep temporary file
   -m*name*   Read macro package **/usr/lib/tmac.***name*
   -n*N*   Number first page of output *N* (default, 1)
   -r*aN*   Set number register *a* to value *N*
   -r*abN*   Set number register *ab* to value *N*; for obvious reasons,
                      ab cannot contain a digit
   -v   Return the number of your version
   -x   Do not eject to bottom of final page

od — Print an octal dump of a file
od [-bcdox] [*file***] [[+]** *offset***[.][b]]**
Options:
   -b   Dump bytes in the default base
   -c   Dump bytes as ASCII characters
   -d   Dump words in decimal
   -o   Dump words in octal
   -x   Dump words in hexadecimal
Default base is octal on the PDP-11; hexadecimal on the i80286, Z-8001,
and M68000. *offset* must be preceded by '+' if *file* is omitted. *offset* is
decimal if '.' is present; 'b' implies 512-byte blocks instead of bytes.

passwd — Set/change login password
passwd [*user***]**

paste — Merge lines of files
paste [-s] [-d *list***]** *file ...*
Options:
   -s   Display lines of input files sequentially across page
   -d *list*   Use *list* as delimiters for output fields

patch — Patch a variable or flag within the kernel
/conf/patch [-k] *image symbol=value ...*
Options:
   -k   Patch the kernel's data memory of the running system
                      via device **/dev/kmem**
   -K   Patch **/dev/kmem** only
   -p   "Peek" — display current values
   -v   Verbose output — display values before and after
                      patching

pathalias — Generate a set of paths among computers
/usr/lib/mail/pathalias [-ivcDf] [-d *link***] [-l** *host***] [-t** *link***] [** *datafile ...* **]**
   -c   Print each path's cost
   -D   Terminal domains
   -d *arg*   **arg** names a dead link
   -f   Cost is that to the first hop in a route, not that of the
                      entire route
   -i   Ignore case of input text
   -l *host*   Set the name of the local host to *host*
   -t *arg*   Write trace information for *arg* onto the standard error
   -v   Verbose: report some statistics on the standard error
                      output.

pathmerge — Merge sorted paths files
/usr/lib/mail/pathmerge *file ...*

pclfont — Prepare a PCL font for downloading via MLP
pclfont [-f *n***]** *font* **[...** *font***]**

phone — Print numbers and addresses from phone directory
phone *person ...*

popd — Pop an item from the directory stack
popd [*item ...* **]**
sh only.

pr — Paginate and print files
pr [*options* **] [** *file …*]
Options:
+*skip*	Skip the first *skip* pages of input before printing
-cols	Print the input in *cols* columns
-h	The next argument is the header (replaces file name)
-l*n*	Set page size to *n* lines (default, 66)
-m	Print each input *file* in a separate column
-n	Number the output lines
-s*c*	Separate each column with character *c*
-t	Suppress top and bottom margins and header
-w*n*	Page width is set to *n* columns (default, 80)

A *file* named '-' means stdin.

prep — Produce a word list
prep [-dfp] [-i *ifile* **] [-o** *ofile* **] [** *file ...* **]**
Options:
-d	Print number of each word output
-f	Fold upper case to lower case
-i *file*	Ignore all words in *file* on output
-o *file*	Output only words from *file*
-p	Print punctuation marks on separate lines (not numbered)

Text is taken from each input *file* or stdin if none. Words consist of
alphabetic characters and apostrophes.

print — Echo text onto the standard output
print [-enru*n***]** [*argument ...*]
Options:
-e	Re-enable expansion of C escape sequences
-n	Don't print newline after list of arguments
-r	Suppress expansion of C escape sequences
-u*n*	Redirect output file descriptor *n*

ksh only.

prof — Print execution profile of a C program
prof [-abcs][*progfile* **[** *monfile* **]]**
Options:
-a	Use all symbols, not just externals
-b	Print all bin information
-c	Print all call information
-s	Report stack usage high water mark

The default *progfile* is **a.out**. The default *monfile* is **mon.out**.

prps — Prepare files for PostScript-compatible printer
prps [*options***] [***file ...* **]**
Options:
-ptsize	Use *ptsize* as the point size (default, 10)
-b	Suppress the box around the page text
-f*font*	Use the given font name (default, Courier)
-F*X*	Use font X, which must be [ABHNPST]
-F*sfx*	Use *sfx* as suffix for font X, which must be **[RBI]**. Default suffixes are "" (R), **-Bold** (B), **-Oblique** (I)
-h	Suppress the header line
-l	Landscape mode (default, portrait)
-l2	Landscape mode, two pages per output page
-n*head*	Use *head* in header line
-p*N*	Print *N* lines of text per output page
-t*N*	Set tab stops to every *N* characters (default, 8)

+*N*   Skip first *N* output pages

ps — Print process status
ps [-][adefglmnrtwx] [-c *sys***] [***mem***] [-p***pid***,***pid***,...,***pid]*
Options:

-a	Print all terminals' processes
-c *sys*	This option does nothing; it is included to preserve the integrity of some shell scripts
-d	Print status of loadable drivers
-e	Same as **-a**; included for compatibility
-f	Put '-' in null fields for placeholders
-g	Give group leader for this process
-k *mem*	Next argument is memory image (default, **/dev/mem**)
-l	Print long format
-m	This option does nothing; it is included to preserve the integrity of some shell scripts
-n	No header line
-p*pid***,***pid***,...,***pid*	Print information for each *pid* in the list
-r	Give the real size of the process
-t	Print CPU times
-w	Wide column format (132 columns instead of default 80)
-x	Print processes with no controlling tty

PSfont — Cook an Adobe font into PostScript format
PSfont [-qs] [*infile***.pfb [** *outfile* **]]**
Options:

-q	Quiet: Suppress warning messages
-s	Suppress **serverdict** header line; to be used when cooking a font to be included within **troff** output

pushd — Push an item onto the directory stack
pushd *[directory0 ... directoryN]*
sh only.

pwd — Print the name of the current directory
pwd

qfind — Quickly find all files with a given name
qfind [-adpv] *name* **...**
qfind [-bv] [-s*directory***]**
Options:

-a	All: search for files or directories
-b	Build file data base
-d	Search for directories only
-p	Partial name matching
-s*directory*	Suppress the reading of *directory* and its children when building the data base
-v	Give verbose output

Run as **root** when using **-b** to find everything.

qpac — Map the file system
qpac *raw_device*

quot — Summarize file-system usage
quot [-c] [-f] [-n] [-t] *filesystem*
Options:

-c	Print file size, number of files of size, and cumulative total blocks up to size
-f	Print number of files plus number of blocks per user
-n	Input (i-number, file system) pairs one per line; output owners and file names (e.g.: **ncheck fs | sort +0n | quot -n fs**)
-t	Print totals (where applicable)

Options **-c** and **-n** are disjoint from other options. Only the superuser **root** can run **quot**.

ranlib — Create index for object library
ranlib *library ...*

read — Assign values to shell variables
read *name ...*
Reads a line from stdin and assign each token to corresponding shell
variable *name*. The shell executes **read** directly.

readonly — Mark a shell variable as read only
readonly

reboot — Reboot the system
/etc/reboot [-p]
Option:
  -p   Prompt user if she really wishes to reboot

ref — Display a C function header
ref *function*

reprint — Reprint a spooled print job
reprint [*job* [*page* [*page*] **]]**

restor — Restore file system
restor *command* **[***dump_device***] [***filesystem***] [***file ...***]**
Options:
  f   Next argument names the dump device
  r   Mass restore (also **R**)
  t   Print taken and since dates of the dump
  v   Verbose (print commentary during mass restore)
  x   Selective extract of argument files (also 'X')

rev — Print text backwards
rev [*file ...***]**

rm — Remove files
rm [-firtv] *file ...*
Options:
  -f   Force: remove unwritable files, suppress error messages
                     and prompts
  -i   Ask before removing each file
  -r   Recursively remove entire directory structure
  -t   Test: perform all checks but do not remove files
  -v   Verbose: report the disposition of each file

rmail — Receive mail from remote sites
rmail [-LlRr] -q *num* **-u** *uuxflags address ...*
Options:
  -L   Send all addresses to the local mailer **lmail** for
                     processing
  -l   Send a domain address to the local mailer **lmail** for
                     processing
  -q *num*   Reset the queueing threshold to *num*
  -R   Reroute paths
  -r   Route the first component of a path in addition to
                     routing domain addresses
  -u *uuxflags* Pass *uuxflags* to **uux**

rmdir — Remove directories
rmdir [-f] *directory ...*
Option:
  -f   Force: remove a file without interactive checking

route — Show or reset a user's default printer
route [*printer***]**

rsmtp — Run batched SMTP mail
/bin/rsmtp

rubik — Play Rubik's cube
/usr/games/rubik

runq — Periodically process the mail queue
/bin/runq

sa — Print a summary of process accounting
sa [-abcjlmnrstu] [-v *N*] [*file*]
Options:
   -a   Commands seen once or unprintable called *****other**
   -b   Sort by average CPU time per call
   -c   Print CPU time as percentage of all CPU time used
   -j   Print average times per call, not totals
   -l   Separate user and system times
   -m   Information per user, not per command
   -n   Sort by number of calls
   -r   Reverse sort
   -s   Condense the information
   -t   Print CPU time as percentage of real time
   -u   Print user and command directly from raw file
   -v*N*   If called no more than *N* times, put it into ****junk****
The default *file* is **/usr/adm/acct**.

savelog — Save a mail log
/usr/lib/mail/savelog [-c *cycle*] [**-g** *group*] [**-l**]
                                  [-m *mode*] [**-u** *user*] [**-t**] *file ...*
   -c *cycle*   Save no more than *cycle* versions of *file*
   -g *group*   Give *group* the group ownership of *file* and its archives
   -l   Do not compress the log files
   -m *mode*   Set permissions on the log files to *mode*
   -t   Touch *file*
   -u *user*   Make *user* the owner of *file*'s archives

scat — Print text files one screenful at a time
scat [[*option ...*] [*file ...*]] ...
Options:
   -1   Do not stop at EOF if exactly one file specified
   -b*n*   Begin output at line *n*
   -c   Mark control characters (overrides **-t**)
   -cs   Like **-c**, but map space to underscore '_', and prefix
                    underscore with '\'
   -ct   Like **-c**, but map tabs to spaces
   -i*n*   Skip *n* columns on output
   -l*n*   Set screen length to *n* lines
   -n   Number input lines
   -r   Remote; no paging
   -s   Squash empty lines
   -S*n*   Seek *n* bytes into input before processing
   -t   Truncate lines to line length (default, wraparound)
   -w*n*   Set screen width to *n* columns
   -x   Expand tabs
Commands:
   <return>   Next page
   <space>   Next line
   /   Next half page
   f   Print file names and line number
   n   Next file
   q   Quit

script — Capture a terminal session into a file
script [-l *logfile*] [*command*]
Options:
   -l   Name *logfile* as the output file

sed — Stream editor
sed [-n] [-e *command*] [**-f** *script*] *... file ...*
Options:
   -e   Direct command follows
   -f   File name of command script follows
   -n   No implicit output

set — Set shell option flags and positional parameters
set [-ceiknstuvx [*name ...***]]** (Bourne shell)
set [[+-]aefhkmnuvx] [[+-]o *name***]** (Korn shell)
Options:

-a	Automatically export all new variables (**ksh**)
-c *string*	Read commands from *string* (**sh**)
-e	Exit on any error
-f	Noglob: Don't expand file names (**ksh**)
-h	Automatically add all commands to hash table (**ksh**)
-i	Shell is always interactive (**sh**)
-k	Place all keyword arguments into environment (**sh**)
-k	Recognize variables anywhere in command (**ksh**)
-m	Enable job control (**ksh**)
-n	Read commands but do not execute
-o *option*	Set *option* (**ksh**)
-s	Read commands from stdin; write to stderr (**sh**)
-t	Read one command rather than entire file (**sh**)
-u	If variable is blank, report error
-v	Print each line as it is read
-x	Print each command as it's executed
-	Cancel **-v** and **-x** options (**sh**)

With **ksh**, prefixing an option with '+' turns it on; prefixing it with '-' turns
it off.

sh — The Bourne shell
sh [-ceiknstuvx] *token ...*
Options:

-c *cmds*	Read commands from *cmds*
-e	Exit on any error if noninteractive
-i	Interactive even if no tty attached
-k	Place all keyword args into global environment
-n	Read commands but do not execute them
-s	Read commands from stdin, write output to stderr
-t	Read and execute one command only
-u	Report error if actual value of shell variable is null
-v	Print each line as read
-x	Print each command and argument as executed
-	Cancel **-v** and **-x** options

The following reserved tokens may not be used in the first position of the
command unless quoted:

case do done elif else fi for in then until while { } ( )

If the first token is not reserved, it is treated as the name of a command.
The remaining tokens are treated as arguments. The characters *** ? [ ]**
specify patterns that match file names. To quote characters or strings,
these escape characters are provided:

**'...' "..." **

Each *token,* unless quoted, is checked for substitutions.

shift — Shift positional parameters
shift
The shell executes **shift** directly.

shutdown — Shut down the system
/etc/shutdown [reboot | halt | single | powerfail] *time*

size — Print size of an object file
size [*file ...***]**

sleep — Stop executing for a specified time
sleep *seconds*

smail — Mail delivery system
smail [*flags* **]** *address ...*

smtpd — SMTP daemon
/bin/smtpd

sort — Sort lines of text
sort [-bcdfimnru] [-t *c*] **[-o** *outfile*] **[-T** *dir*] **[+**beg**[-**end**]][**file …**]**
Options:

-c	Check if input is already ordered
-m	Merge already-sorted input files
-o *name*	Place output in *name*, not stdout
-tc	Tab character is *c*
-T *dir*	Use *dir* for temporary files
-u	Output only records with unique keys

Key options:

-b	Skip leading blanks in fields
-d	Dictionary ordering for keys
-f	Fold upper case into lower case in key comparison
-i	Ignore control characters in key comparison
-n	Numeric comparison
-r	Reverse sort ordering

Position:

+m.nf	Key starts *m* fields into record and *n* characters into that field; *f* may contain optional flags from key options above which apply only to that positional
-m.nf	Optional ending position of key (same form as above)

If no *file* is given, sort stdin.

spac — Sort a file system
spac *raw_device*

spell — Find spelling errors
spell [-a][-b][file ...**]**
Options:

-a	Use American variant of the dictionary (default)
-b	Use British variant of the dictionary

split — Split a text file into smaller files
split [-lines**][-c**count**][**infile **[**outfile**]]**
If *infile is '-' or no infile*, stdin is read. *outfile* defaults to **x**. *lines* is number of lines for text files, *count* is the character count for binary files.

srcpath — Find source files
srcpath [-aw] [-p *path*] *filename pattern* **...**
Options:

-a	Disable shadowing: print all instances of file it finds along **SRCPATH**, not just first
-p *path*	Use *path* instead of **SRCPATH**
-w	Print warning when you lack permission to read file or directory

strings — Print all character strings from a file
strings [-dopx] [-length**] [**file ... **]**
Options:

-d	Print offset of each string in decimal
-o	Print offset of each string in octal
-p	Mask out the parity bit
-x	Print offset of each string in hexadecimal

strip — Strip tables from executable file
strip *file [...]*

stty — Set/print terminal modes
stty
stty -a
stty -g
stty *x:x: ... :x*
stty *arglist ...*
Common Options:

cooked	Insert device into cooked mode
eof *c*	Set the end-of-file character to *c*
eol *c*	Set the end-of-line character to *c*
erase *c*	Set the erase character to *c*
evenp	Turn on even parity

intr *c*	Set the interrupt character to *c*
kill *c*	Set the kill character to *c*
lcase	Map upper-case characters to lower case
-lcase	Turn off mapping of upper-case characters to lower case
oddp	Turn on odd parity.
parity	Turn on even parity
-parity	Turn off parity
quit *c*	Set the quit character to *c*
raw	Insert device into raw mode
sane	Restore a device to sanity, e.g., after an editor or communications program died unexpectedly

If no *option* is specified, **stty** prints the modes of the standard-input device on stderr.

su — Substitute user id, become superuser
su [*user* [*command*]**]**

sum — Print checksum of a file
sum [*file ...***]**

sync — Flush system buffers
sync

tail — Print the end of a file
tail [+*n***[bcfl]] [***file***]**
tail [-*n***[bcfl]] [***file***]**

Options:

+*n*	*n* counts from beginning of file
-*n*	*n* counts from end of file
b	*n* is in blocks
c	*n* is in characters (bytes)
f	Open tail of file, then display new material as it is added to the file. File remains open until you type interrupt (usually **<ctrl-C>**).
l	*n* is in lines (default)

tape — Manipulate a tape device
tape command [*count***] [***device***]**

Commands:

erase	Erase the tape. SCSI tape only.
retension	Retension the tape.
rewind	Rewind the tape.
rfm	Move the tape forward to the next file mark; in effect, skip the current file. SCSI tape only.
seek *position*	Move the tape device's seek point to byte *position*. Floppy tape only.
status	Read the status of the tape device. Floppy tape only.
tell	Display the byte position at which the next read or write operation will be performed. Floppy tape only.

tar — Archiving/backup utility
tar [*options***]** *file ...*
tar is now a link to the command **gtar**. See its Lexicon entry for details.

tee — Copy input to multiple output streams
tee [-a] [-i] [*file ...***]**

Options:

-a	Append to each output *file*
-i	Ignore interrupts

test — Evaluate conditional expression
test *expression ...*

Options:

! *exp*	Negate logical value of *exp*
string1 **!=** *string2*	*string1* does not equal *string2*
string1 **<** *string2*	*string1* is lexicographically less than *string2* (**sh** only)
string1 **=** *string2*	*string1* equals *string2*

string1 **>** *string2*	
	string1 is lexicographically greater than *string2* (**sh** only)
(*exp***)**	Parentheses group expressions
exp1 **-a** *exp2*	Both expressions are true
-b *file*	*file* is block-special
-c *file*	*file* is character-special
-d *file*	*file* is a directory
-e *file*	*file* exists (**sh** only)
file1 **-ef** *file2*	Files are identical
n1 **-eq** *n2*	Numbers are equal
-f *file*	*file* exists and is an ordinary file
-g *file*	*file* has **setgid** bit set
n1 **-ge** *n2*	*n1* is greater than or equal to *n2*
n1 **-gt** *n2*	*n1* greater than *n2*
-k *file*	*file* has sticky bit set
-L *file*	*file* is a symbolic link
n1 **-le** *n2*	*n1* is less than or equal to *n2*
n1 **-lt** *n2*	*n1* is less than *n2*
-n *string*	**string** has non-zero length
n1 **-ne** *n2*	*n1* does not equal *n2*
f1 **-nt** *f2*	*f1* is newer than *f2*
exp1 **-o** *exp2*	Either *exp1* or *exp2* is true
f1 **-ot** *f2*	*f1* is older than *f2*
-p *file*	*file* is a named pipe
-r *file*	*file* is readable
-s *file*	*file* exists and has nonzero length
-t *[fd]*	*fd* describes a terminal
-u *file*	*file* has **setuid** set
-w *file*	*file* is writable
-x *file*	*file* is executable
-z *string*	*string* has zero length
string	*string* has non-zero length

tic — Compile a terminfo description
tic [-v[*n***]]** *sourcefile*

Option:
    -v   Verbose: Include debugging and tracing information.

time — Time the execution of a command
time [*command***]**

times — Print total user and system times
times

touch — Update modification time of a file
touch [-c] *file ...*

Option:
    -c   Do not create *file* if it does not exist
The shell executes **touch** directly.

tr — Translate characters
tr [-cds] *string1* **[***string2***]**

Options:
    -c   Complement the characters in *string1*
    -d   Delete characters found in *string1 (no string2* needed)
    -s   Squeeze multiple output mappings onto one character
Both strings may contain ranges. Characters may have form *nnn*.

trap — Execute command on receipt of signal
trap [*command***] [***n ...***]**
The shell executes *command* on receipt of signal *n ....* If *command* omitted,
the shell resets traps on given signals to original values. If *command* is a
null string, given signals are ignored. If *n* is zero, the shell executes
command when it exits. With no arguments, it prints currently set traps.
The shell executes trap directly.

troff — Extended text-formatting language
troff [*option ...***] [***file ...***]**

Options:
    -D   Display available fonts

-f *name*	Write temporary file in file *name*
-i	Read stdin after each *file* has been read
-k	Keep temporary file
-l	Landscape mode
-mname	Read macro package **/usr/lib/tmac.***name*
-nN	Number first page of output *N* (default, one)
-p	Produce PostScript output
-raN	Set number register *a* to value *N*
-rabN	Set number register *ab* to value *N*; for obvious reasons, *ab* cannot contain a digit
-v	Return the number of your version
-x	Do not eject to bottom of final page

true — Unconditional success
true

tsort — Topological sort
tsort [*file***]**

ttt — Play 3-D tic-tac-toe
/usr/games/ttt

tty — Print the user's terminal name
tty

ttystat — Get terminal status
/etc/ttystat [-d] *port*
Option:
   -d   Print status of *port*
Returns exit status 1 if specified port is enabled, 0 if disabled. Prints
nothing unless **-d** option specified.

ttytype — Select a default terminal type for a port
ttytype

typeset — Set/list variables and their attributes
typeset
typeset [+-]fr
typeset [irx] *variable***=***value*
First form: List all variables and their attributes
Second form:
+f	List functions
-f	List functions plus values
+r	List read-only variables
-r	List read-only variables plus values

Third form: Set *variable* to equal *value*
i	Store *value* as an integer
r	List read-only variables
x	Export *variable*=*fR* to environment

ksh only.

typo — Detect possible typographical and spelling errors
typo [-nrs][*file ...***]**
Options:
-n	Do not use built-in English statistics or dictionary
-r	Raw; do not remove nroff commands from the input
-s	Produce *digrams* and *trigrams* files (maintenance only)

umask — Set the file-creation mask
umask [*OOO***]**
OOO are three octal digits that describe file permissions. The shell
executes **umask** directly.

umount — Unmount file system
/etc/umount *special*

unalias — Remove an alias
unalias *alias ...*

uname — Print information about
uname [-amnrsv]
uname [-S *systemname* **]**

uncompress — Uncompress a compressed file
uncompress [*file ...* **]**

uniq — Remove/count repeated lines in a sorted file
uniq [-cdu] [-*n*] [+*n*] [*infile*[*outfile*]]
Options:
   -c   Print duplication count with lines
   -d   Print only duplicated lines
   -n   Skip *n* fields during comparison
   +n   Skip *n* characters (after skipping fields)
   -u   Print only non-repeated lines

units — Convert measurements
units [-u]
Option:
   -u   Update binary file only
units works interactively.

unpack — GNU utility to uncompress files
unpack [-cfhLrtvV] [*file ...* **]**
Options:
   -c   Write output to the standard output
   -f   Force: decompress a *file* even if its has multiple links or
               the corresponding file already exists
   -h   Help: display a screenful of information about how to
               use this command
   -L   Display licensing information
   -r   Recurse: if a *file* is a directory, attempt to uncompress
               all files within it
   -t   Test the integrity of a compressed *file*
   -v   Give a verbose description of each action
   -V   Print the current version of this command

unset — Unset an environment variable or shell function
unset *environmental_variable*
unset -f *shell_function*

until — Execute commands repeatedly
until *sequence1* **[do** *sequence2* **] done**
Both **do** and **done** must be the first token on a line or preceded by ';'. The
shell executes **until** directly.

unzip — Un-zip a zipped archive
unzip *archive* **[-cfpux** *file ...*] **[-ltvz] [-anojqUV]**
Options
   -c [*file ...*]   Extract files, write them to the standard output
   -f [*file ...*]   Update existing files from the archive; do not create new
               files
   -l   List the contents of the archive, short format
   -p [*file ... | command*]
               Extract to pipe, no messages
   -t   Test archive integrity
   -u [*file ...*]   Update files, create if necessary
   -v   List files (verbose format)
   -x [*file ...*]   Extract files (default)
   -z   Display archive's comment
Modifiers
   -a   Convert text from format to format
   -j   Ignore ("junk") paths; do not make directories
   -n   Never overwrite existing files
   -o   Overwrite files without prompting
   -q   Quiet mode
   -qq   Quieter mode
   -U   Do not convert file names to lower-case letters

upac — De-fragment a file system without sorting
upac *raw_device*

uuchk — Check UUCP configuration
/usr/lib/uucp/uuchk [-I*file***] [v] [--help]**
   -I*file*   Use *file* instead of standard configuration files
   -v   Display version of program
   --help   Print a help message

uucico — Communicate with a remote site
/usr/lib/uucp/uucico [-D] [-c*site***] [-I***file***] [-p***port***] [-r0] [-r1] [-s***site***] [-S***site***] [-x***level***]**
Options:
   -c*site*   "Cron" mode: Do not log an error message should a call
                 conflict with a legal calling time
   -D   Do not detach from the device until the contact with the
                 remote system concludes
   -e   Force **uucico** to produce its own **Password:** prompt
   -I *file*   Read configuration information from *file*, instead of from
                 the default file **/usr/lib/uucp/sys**
   -l   Force **uucico** to produce its own **login:** prompt
   -p*port*   Use *port*. When used with the options **-s** or **-S**, dial out
                 on *port*. When used with option **-e**, listen to *port*
   -q   Quiet: do not invoke daemon **uuxqt** on the remote
                 system
   -r0   Act as slave in polling process
   -r1   Act as master in polling process; default
   -s*site*   Name *site* as a place to be polled
   -S*site*   Force the system to poll *site* immediately, if the present
                 time lies within the legal times for *site*
   -w   Begin an endless loop of prompts, as with the **-e** option,
                 after contacting a system with the options **-s** or **-S**
   -x*activity*   Log a given *activity*, for debugging purposes

uuconv — Convert UUCP configuration files to Taylor format
/usr/lib/uucp/uuconv -i *input* **-o** *output* **[-p** *program***] [-I** *file***]**

uucp — Spool files for transmission to other systems
uucp [-cCdfmr] [-n*user***] [-x***activity***]** *source ... dest*
Options:
   -c   Do not copy source to spool directory; rather use the file
                 itself
   -C   Copy source file to spool directory (default; same as **-p**)
   -d   Create directories as required on destination
   -f   Do not make intermediate directories; fail if they do not
                 exist
   -I *file*   Use *file* as configuration file
   -j   Report job's process identifier
   -m   Send mail to requester when file is sent
   -n*user*   Notify *user* (on destination system) when file received
   -p   Copy source file to spool directory (default; same as **-C**)
   -R   Copy directories recursively
   -r   Spool transfer request, do not initiate **uucico**
   -s *file*   Write status upon completion of job into *file*
   -u *user*   Set user name to *user*
   -W   Do not add the current directory to file names on the
                 remote system
   -x*activity*   Log a given *activity*, for debugging purposes

uudecode — Decode a binary file sent from a remote system
uudecode [*file* **]**

uuencode — Encode a binary file for transmission
uuencode [*source* **]** *file_label* **[<** *source* **] >** *output*

uuinstall — Install or modify
uuinstall

uulog — Read a log
uulog [-f*system***] [-s***system***] [-n***number***] [-x**]
Options:
-f*system*	Show activity as it is logged; like **tail -f**
-n*number*	Display *number* lines from the end of the log
-s*system*	Display the log for *system*
-x	Display logs for **uuxqt** instead of **uucico**

uumkdir — Create UUCP directories
/usr/lib/uucp/uumkdir [-m *mode* **] [-p]** *directory ...*

uumvlog — Archive log files
uumvlog *days*
Options:
days	Number of days for which logs should be kept

uuname — List names of known systems
uuname [-l]
Option:
-l	Print name of the local system

uupick — Pick up a file uploaded from a remote system
/usr/bin/uupick [-s *system***] [-I** *file***] [-x** *event***]** *file ...*

uurmlock — Remove lock files
uurmlock

uusched — Call all systems that have jobs waiting for them
/usr/lib/uucp/uusched

uustat — UUCP status inquiry and control
uustat [-eKiMNQ] [-B *lines***] [-cC** *command***] [-o** *hours***] [-sS** *system***] [-uU**
*user***] [-y** *hours***]**
uustat -a
uustat [-k *jobid***] [-r** *jobid***]**
uustat -m
uustat -p
uustat -q
Options:
-a	List queued requests to transfer files
-C *command*	List all jobs except those requesting execution of *command*
-c *command*	List every job that requests the execution of *command*
-e	List requests queued to execute a program on a remote system
-I *file*	Read configuration information from *file*
-i	For each listed job, prompt whether to kill the job
-K	Automatically kill each listed job
-k *jobid*	Kill the job with the identifier *jobid*
-M	For each job, send mail to administrator
-m	Display status of conversations for all remote systems
-N	For each job, send mail to user who requested job
-o *hours*	List all jobs that have been queued longer than *hours*
-p	Display status of processes holding locks
-Q	Work quietly
-q	Display status of commands, executions, and conversations for all remote systems
-r *jobid*	Rejuvinate the job with job identifier *jobid*
-S *system*	List all jobs except ones queued for *system*
-s *system*	List every job queued for *system*
-U *user*	List all jobs except ones queued for *user*
-u *user*	List every job queued for *user*
-W	Specify comment to include in mail sent with options **-M** or **-N**
-x *type*	Turn on particular types of debugging
-y *hours*	List all jobs queued less than *hours*

uuto — Send a file to a remote system
/usr/bin/uuto *file ... file remote_system*

uutouch — Touch a file to trigger poll
uutouch *system*

uutry — Debugging script for
uutry *remotesystem* **[-x***debuglevel***]**"

uux — Execute a command on a remote system
uux [-a *user***] [-rnpz]** *command-string*

Options:
-a *user*	Name *user* as requester
-C	Copy local files to the spool directory
-c	Do not copy local files to the spool directory
-g *l*	Set importance of transmission; *l* is a single ASCII character
-I *file*	Read configuration information from *file*
-j	Print job identifiers on standard output
-l	Link local files into spool directory
-n	Suppress notification of command failure
-p	Input to **uux** is a pipe or input redirection
-r	Queue **uux** request; do not invoke **uucico**
-x *event*	Log each *event* in the execution of **uux**
-z	Notify requestor when *command-line* succeeds

uuxqt — Execute commands requested by a remote system
uuxqt

vi — Clone of Berkeley-style screen editor
vi [*options* **] [+***cmd* **] [** *file1 ... file27* **]**

Options:
-e	Begin in colon-command mode
-i	Begin in input mode
-r	Recover a previous edit
-R	Read-only mode
-t *tag*	Begin editing at *tag*
-m	Use in error-handling mode
-v	Begin in visual-command mode
+*command*	Execute *command* before editing

view — Screen-oriented viewing utility
view *file1 ... file27*

vsh — Interactive graphical shell
vsh [-d*directory***] [-eirt]**
Options:
-d*directory*	Begin work *directory* rather than in the directory from which you exited **vsh**; if no *directory* named, begin work in the current directory
-e	Do not use VT-100 graphics-character set
-i	Restrict the use of the **Install** menu
-r	Restrict user's ability to customize **vsh** or access a shell
-t	Use full VT-100 graphics mapping

wait — Await completion of background process
wait [*pid***]**
pid identifies the process whose completion is awaited. If no *pid* is given,
wait awaits completion of all background processes. The shell executes
wait directly.

wall — Send a message to all logged-in users
/etc/wall

wc — Count words, lines, and characters in text files
wc [-clw] [*file...***]**
Options:
-c	Print count of characters
-l	Print count of lines
-w	Print count of words

If no *file* is given, **wc** reads stdin; if more than one *file* is given, it also prints a total.

whence — List a command's type
whence [-v] *command ...*
ksh only.

whereis — Locate source, binary, and manual files
whereis [-bmrsu] [-BMS *dir ...* **-f]** *name ...*
Options:
-b	Search only for for binary files
-m	Search only for manual pages
-r	Search each *dir* downwardly recursively
-s	Search only for source files
-u	Search for unusual files
-B	Search each *dir* for binary files
-M	Search each *dir* for manual pages
-R	Search each *dir* downwardly recursively
-S	Search each *dir* for source files
-f	Terminate directory list begun by **-BMRS** options

which — Locate executable files
which *command ...*

while — Execute commands repeatedly
while *sequence1* **[do** *sequence2***] done**
Both **do** and **done** must be the first token on a line or preceded by ';'. The shell executes **while** directly.

who — Print who is logged in
who [*file***] [am i]**

write — Converse with another user
write *user* **[** *tty* **]**
Name the *tty* if *user* is logged in on more than one port.

xargs — Execute a command with many arguments
xargs *command argument ... argument*

yacc — Parser generator
yacc [*option ...***]** *file*
cc y.tab.c [-ly]
Options:
-d	Enable debugging output (implies **-v**)
-hdr	Next argument is name of header file (default, **y.tab.h**)
-items	**Allow** *N* items per state.
-l	Next argument is name of listfile (default, **y.output**)
sprod *N*	Allow *N* symbols per production (default, 20)
-st	Print statistics on standard output
-v	Verbose (extra output in listfile)

After each of the following, the next argument is a number to reset table size:
-nterms	Nonterminal symbols (default, 100)
-prods	Productions or rules (default, 350)
-states	States (default, 300)
-terms	Terminal symbols (default, 300)
-types	Types (default, 10)

yes — Print infinitely many responses
yes [*string* **]**

zcat — Concatenate a compressed file
zcat [*file***[.Z|.gz]** ... **]**

zcmp — Compare compressed files
zcmp [-ls] *file1***[.gz]** *file2***[.gz] [***skip1 skip2***]**
Options:
-l	Print byte number and bytes at each difference
-s	Return status (print nothing)

If *file1* is '-', use stdin. If *skip1* and *skip2* are present, they are the number

of bytes to skip before comparing *file1* and *file2*, respectively.

zdiff — Compare two compressed files
zdiff [-bdefh] [-c *symbol***]** *file1 file2*
Options:
-b	Ignore trailing blanks; all strings of blanks are equal
-c *sym*	Make **cpp** input conditionalized on *sym*
-d	Use **-h** algorithm for large (>25,000 character) files
-e	Make **ed** script
-f	Make fake (non-usable) **ed** script
-h	Half-hearted algorithm (works on long files)
-s	Make **sed** script

If either *file1* or *file2* is '-', stdin is used. If one *file* is a directory, the other *file* under that directory is used.

zforce — Force the suffix .gz onto every gzip file
zforce [*file …* **]**

zgrep — Search compressed files for a regular expression
zgrep [-abcefhilnsvxy] [*pattern***] [***file …***]**
Options:
-a	Extra metacharacters supported ('(…)', '|', '+', and '?')
-b	Each output line has block number of match
-c	Print only count of matching lines
-e	Next argument is pattern
-f	Next argument is file containing one pattern per line
-h	Suppress printing of file names on matched lines
-i	Ignore case when matching letters in *pattern*
-l	Print only names of files containing matches
-n	Print line number of file with each matched line output
-s	Suppress output, just return status
-v	Negate sense of match
-x	Exact match (don't expand metacharacters)
-y	Lower-case letters in *pattern* match only upper-case

The *pattern* is a regular expression roughly like that found in **ed**. If no *file* is specified, stdin is read.

zip — Zip files into a compressed archive
zip [-*options***] [-b** *path***] [-t** *mmddyy***]** *zipfile file … ***[-x** *file …***]**
Options:
-b *path*	Use *path* for temporary files
-c	Add one-line comments to the archive
-d	Delete each "file" from the archive
-e	Encrypt the file
-ee	Verify the encryption key
-f	"Freshen" the contents of the archive the archive.
-g	"Grow" — that is, append files to — an existing archive
-h	Display a help message
-i	Only implode the files
-j	"Junk" (that is, do not record) directory names
-k	Mimic a PKZIP-made zipfile
-l	Show the software license
-m	Delete each file
-n	Do not compress special suffixes
-o	Make zipfile as old as latest entry
-q	Operate quietly
-r	Operate recursively
-s	Compress files only
-t	Manipulate only the files updated since "mmddyy"
-u	Update only
-x	Exclude each "file" from those manipulated
-z	Add a zipfile comment
-0	Use level-0 compression
-9	Use level-9 compression

zmore — Display compressed text one page at a time
zmore [-cdflsu] [-*window_size* **] [+***line_number* **] [+/***pattern* **] [***file …* **] [-]**
Options:
-	Read/display stdin
-c	Paint screen from top down
-d	Prompt user to quit after each screenful of text

-f	Count lines from file rather than screen-display lines
-l	Do not treat **<ctrl-L>** as special
-s	Squeeze consecutive blank lines into one
-u	Display backspaces as control characters
+line_number	
	Begin display at *line_number*
+/pattern	Begin display at first line to contain *pattern*

znew — Recompress .Z files to .gz files
znew [-ftv9PK] [*file*.**Z ...]**

Options:

-f	Force recompression
-t	Test the new files before deleting originals
-v	Verbose mode
-9	Use the slowest, most thorough compression method (optimal compression)
-P	Use pipes for the conversion, to reduce disk space usage
-K	Keep the **.Z** file when it is smaller than the **.gz** file

vi Commands

vi is a modal editor whose command structure resembles the **ed** line editor. *Modal* means that a keystroke assumes a different meaning, depending upon mode that the editor is in. **vi** uses three modes: *visual-command mode, colon-command mode,* and *input mode.* To tell visual-command mode from input mode, press **<esc>** key. If **vi** beeps, you are in visual-command mode; otherwise, you were in input mode, but pressing **<esc>** switched you to visual-command mode. Entering command **<esc>:set showmode** causes **vi** to display the current mode at the bottom of your screen.

Visual-Mode Cursor Movement Commands

count indicates that the command can be prefaced by an argument that tells **vi** how often to execute it. *move* means that the command can be followed by a movement command, after which the command is executed on text that lies between the point where command was first typed and the point to which the cursor was moved. Typing command a second time executes the command for the entire line upon which cursor is positioned. *key* means that the command must be followed by an argument. Note that in the following sections "up" indicates towards the beginning of the file, and "down" indicates towards the end of the file.

[count] **<ctrl-B>**	Move up by one screenful.
[count] **<ctrl-D>**	Scroll down *count* lines (default, one-half screenful).
[count] **<ctrl-E>**	Scroll down *count* lines (default, one).
<ctrl-F>	Move down by one screenful.
<ctrl-G>	Show file status and current line.
[count] **<ctrl-H>**	Move one character to the left.
[count] **<ctrl-J>**	Move down *count* lines.
<ctrl-L>	Redraw screen.
[count] **<ctrl-M>**	
	Move to beginning of next line.
[count] **<ctrl-N>**	Move down *count* lines (default, one).
[count] **<ctrl-P>**	Move up *count* lines (default, one).
<ctrl-R>	Redraw screen.
[count] **<ctrl-U>**	Scroll up *count* lines (default, one-half screenful).
[count] **<ctrl-Y>**	Scroll up *count* lines (default, one).
<ctrl-]>	If cursor is on a tag name, go to that tag.
<ctrl-^>	Switch to previous file.
[count] **<space>**	Move right *count* spaces (default, one).
! *[move]*	Run selected text through external filter program.
key	Select which cut buffer to use next.
$	Move to end of current line.
%	Move to matching (){}[] character.
' *key*	Move to a marked line.
' *key*	Move to *key* character.
[count] **(**	Move backward *count* sentences (default, one).
[count] **)**	Move forward *count* sentences (default, one).
*	Go to next error in error list.
[count] **+**	Move to beginning of next line.
[count] **,**	Repeat previous **f** or **t** command, but move in opposite direction.
[count] **-**	Move to beginning of preceding line.
[count] **.**	Repeat previous *edit* command.
/ *text*	Search forward for *text,* which can be a regular expression.
0	If not part of a count, move to first character of this line.
:	Switch to colon-command mode to execute one command.
[count] **;**	Repeat previous **f** or **t** command.
? *text*	Search backwards for *text,* which can be a regular expression.
@ *key*	Execute contents of a cut-buffer as **vi** commands.
[count] **B**	Move backwards *count* words (default, one).
[count] **E**	Move forwards to end of *count*'th space-delimited word (default, one).
[count] **F** *key*	Move left to *count*'th occurrence of given character (default, first).
[count] **G**	Move to *count*'th line in file (default, last).
[count] **H**	Move *count* lines down from top of screen (default, top).
K	Look up a keyword.
[count] **L**	Move *count* lines up from bottom of screen (default,

	bottom).
M	Move to middle of screen.
N	Repeat last search, but in opposite direction.
P	Paste text before cursor.
Q	Shift to colon-command mode.
[count] **T** *key*	Move left *almost* to given character.
U	Undo all recent changes to current line.
[count] **U**	Move forward *count* words (default, one).
[count] **W**	Move forwards to beginning of *count*'th space-delimited word (default, one).
[count] **Y**	Copy (or "yank") *count* lines into a cut buffer (default, one).
Z Z	Save file and exit.
[[	Move back one section.
]]	Move forward one section.
^	Move to beginning of current line, but after indent.
[count] **b**	Move back *count* words.
[count] **e**	Move forward to end of *count*'th word.
[count] **f** *key*	Move rightward to *count*'th occurrence of given character.
[count] **h**	Move left *count* characters (default, one).
[count] **j**	Move down *count* characters (default, one).
[count] **k**	Move up *count* characters (default, one).
[count] **l**	Move right *count* characters (default, one).
m *key*	Mark a line or character.
n	Repeat previous search.
p	Paste text after cursor.
[count] **t** *key*	Move rightward *almost* to *count*'th occurrence of given character (default, one).
u	Undo previous edit command.
[count] **w**	Move forward *count* words (default, one).
y *move*	Copy (or "yank") text into a cut buffer.
z *key*	Scroll screen, repositioning current line as follows: **+** indicates top of screen, indicates bottom, **.** indicates middle.
[count] **{**	Move back *count* paragraphs (default, one).
[count] **l**	Move to *count*'th column on screen (leftmost, one).
[count] **}**	Move forward *count* paragraphs (default, one).

Visual-Mode Editing Commands

[count] **##**	Increment a number by *count* (default, one).
[count] **&**	Repeat previous **:s//** command.
< *move*	Shift enclosed text left.
> *move*	Shift enclosed text right.
[count] **A** *input*	Append input at end of line.
C *input*	Change text from cursor through end of line.
D	Delete text from cursor through end of line.
[count] **I** *input*	Insert text at beginning of line (after indentations).
[count] **J**	Join lines current with following line.
[count] **O** *input*	Open a new line above current line.
R *input*	Overtype.
[count] **S** input	Change lines, like **cc**.
[count] **X**	Delete *count* characters from left of cursor (default, one).
[count] **a** *input*	Insert text after the cursor.
c *move*	Change text.
d *move*	Delete text.
[count] **i** *input*	Insert text at cursor.
[count] **o** *input*	Open a new line below current line.
[count] **r** *key*	Replace *count* characters with text you type (default, one).
[count] **s** *input*	Replace *count* characters with text you type (default, one).
[count] **x**	Delete character at which cursor is positioned.
[count] **~**	Toggle a character between upper case and lower case.

Colon-Mode Commands

Colon-mode commands come from the **ex** line editor. *line* is a regular expression that indicates whether the command is executed on one or more lines. Some commands can be used with an optional exclamation point, to suppress normal warnings and prompts.

abbr *[word full_form]*	
	Define *word* as an abbreviation for *full_form*.
[line] **append**	Insert text after current line.
args *[file1 ... fileN]*	

With no arguments, print files list on **vi**'s command line. With one or more arguments, change name of current file.

cc *[files]*   Invoke C compiler to compile *files*, and redirect all error messages into file **errlist**. After compiler exits, scan **errlist** for error messages; if one is found, jump to line and file indicated on error line, and display error message on status line.

cd *[directory]*   Switch current working directory. With no argument, switch to **$HOME** directory.

[line][,line] **change** *[x]*
Replace range of lines with contents of cut-buffer *x*.

chdir *[directory]*
Same as **cd** command.

[line][,line] **copy** *targetline*
Copy range of lines to after *targetline*.

[line][,line] **delete** *[x]*
Move range of lines into cut buffer *x*.

digraph*[!] [XX [Y]]*
Set *XX* as a digraph for *Y*. With no arguments, display all currently defined digraphs. With one argument, undefine *XX* as a digraph.

edit*[!] [file]*   Edit a file not named on **vi** command line.

errlist*[!] [errlist]*Find next error message in file **errlist**, as generated through **vi**'s **cc** or **make** commands.

file *[file]*   With an argument, change the output file to *file*. Without an argument, print information about current output file.

[line][,line] **global** */regexp/ command*
Search range of lines for all lines that contain regular expression *regexp*, and execute *command* upon each.

[line] **insert**   Insert text before current line.

[line][,line] **join** Concatenate range of lines into one line.

[line][,line] **list** Display requested range of lines, making all embedded control characters explicit.

make *[target]* Same as **cc** command, except that **make** is executed.

map*[!] key mapped_to*
Remap *key* to *mapped_to*. Normally, remapping applies just to visual-command mode; '!' tells **vi** to remap key under all modes. With no arguments, show all current key mappings.

[line] **mark** *x* Set a mark on *line*, and name it *x*.

mkexrc   Save current configuration into file **./.exrc**, which will be read next time you invoke **vi**.

[line][,line] **move** *targetline*
Move range of lines to after *targetline*.

next*[!] [files]* Switch to next file on **vi** command line.

Next*[!]*   Switch to preceeding file on **vi** command line.

[line][,line] **number**
Display the range of lines, with line numbers.

previous*[!]*   Switch to preceeding file on **vi** command line.

[line][,line] **print**Display specified range of lines.

[line] **put** *[x]*   Copy text from cut buffer *x* after current line.

quit*[!]*   Quit **vi**, and return to the shell.

[line] **read** *file* Read *file* and insert contents after **line** (default, the last line).

rewind*[!]*   Switch to first file on **vi** command line.

set *[options]* Set a **vi** option. With no arguments, list options which are set. Command **set all** displays all possible options.

shell   Invoke a shell.

source *file* Read a set of colon-mode commands from *file*, and execute them.

[line][,line] **substitute** */regexp/replacement/[p][g][c]*
For the range of lines, replace first instance of *regexp* with *replacement*. *p* tells **vi** to print *last* line upon which a substitution was performed. *g* means perform a global substitution, i.e., replace all instances of *regexp* on each line with *replacement*. *c* tells **vi** to ask for confirmation before performing each substitution.

tag*[!] tagname* Find *tagname* in file **tags**, which records information about all tags. If found, jump to file and line upon which tag is set.

[line][,line] **to** *targetline*
                Copy range of lines to after *targetline*.
unabbr *word*   Unabbreviate *word*.
undo   Undo last editing command.
unmap*[!] key*   Unmap *key*.
version   Display current version of **vi**.
[line][,line] **vglobal** */regexp/ command*
                Search range of lines for all lines that do not contain
                regular expression *regexp*, and execute *command* upon
                each.
visual   Enter visual-command mode.
wq   Save changed file, and exit.
[line][,line] **write***[!] [[>>][file]*
                Write file being edited into *file*. With >> argument,
                append edited text onto end of *file*.
xit*[!]*   Same as **wq** command, described above, except that it
                does not write files that have not changed.
[line][,line] **yank** *[x]*
                Copy range of lines into cut buffer *x*.
[line][,line] **!** *command*
                Execute *command* under a subshell, then return.
[line][,line] **<**   Shift range of lines left by one tabwidth.
[line][,line] **=**   With no range of lines specified, print number of current
                line. With line arguments, print the endpoints of lines
                in question, and number of lines that lie between them.
                (Remember, *line* can be a regular expression as well as a
                number.)
[line][,line] **>**   Shift range of lines right by one tabwidth.
[line][,line] **&**   Repeat last substitution command.
@ x   Read contents of cut-buffer *x* as a set of colon-mode
                commands, and execute them.

Input-Mode Commands

In input mode, most keystrokes are inserted directly into the text; however,
the following keystrokes are still interpreted as commands.

<ctrl-A>   Insert a copy of last input text.
<ctrl-D>   Delete one indent character.
<ctrl-H>   Erase character before cursor.
<ctrl-L>   Redraw screen.
<ctrl-M>   Insert a newline.
<ctrl-P>   Insert contents of cut buffer.
<ctrl-R>   Redraw screen, like **<ctrl-L>**.
<ctrl-T>   Insert an indent character.
<ctrl-U>   Move to beginning of line.
<ctrl-V>   Insert following keystroke, even if special.
<ctrl-W>   Backspace to beginning of current word.
<ctrl-Z><ctrl-Z>
                Write file and exit **vi**.
<esc>   Shift from input mode to visual-command mode.
<del>   Delete current character.

MicroEMACS Commands

Moving the Cursor

\<ctrl-A\>	Move to start of line.
\<ctrl-B\>	Move backward by characters.
\<esc\>B	Move backward by words.
\<ctrl-E\>	Move to end of line.
\<ctrl-F\>	Move forward by characters.
\<esc\>F	Move forward by words.
\<esc\>G	Go to an absolute line number in a file. Same as **\<ctrl-X\>G**.
\<ctrl-N\>	Move to next line.
\<ctrl-P\>	Move to previous line.
\<ctrl-V\>	Move forward by page.
\<esc\>V	Move backward by pages.
\<ctrl-X\>=	Print the current position.
\<ctrl-X\>G	Go to an absolute line number in a file Can be used with an argument; otherwise, it prompts for line number. Same as **\<esc\>G**.
\<ctrl-X\>[	Go to matching C delimiter. MicroEMACS recognizes the delimiters [,], {, }, (,), /*, and */.
\<ctrl-X\>]	Toggle reverse-video display of matching C delimiters.
\<esc\>!	Move current line to line within window given by *argument*; position is in lines from the top if positive, in lines from bottom if negative, and center of window if zero.
\<esc\>\<	Move to beginning of current buffer.
\<esc\>\>	Move to the end of the current buffer.

Killing and Deleting

\<ctrl-D\>	Delete next character.
\<esc\>D	Kill next word.
\<ctrl-H\>	If no argument, delete previous character.
\<ctrl-K\>	With no argument, kill from current position to end of line; if at end, kill newline. With argument set to one, kill from beginning of line to current position. Otherwise, kill *argument* lines forward (if positive) or backward (if negative).
\<ctrl-W\>	Kill text from current position to mark.
\<ctrl-X\>\<ctrl-O\>	
	Kill blank lines at current position.
\<ctrl-Y\>	Copy the kill buffer into text at current position; set current position to end of new text.
\<esc\>\<ctrl-H\>	Kill previous word.
\<esc\>\<DEL\>	Kill previous word.
\<DEL\>	If no argument, delete previous character. Otherwise, kill *argument* previous characters.

Windows

\<ctrl-X\>1	Display only current window.
\<ctrl-X\>2	Split current window into two windows. This command is usually followed by **\<ctrl-X\>B** or **\<ctrl-X\>\<ctrl-V\>**.
\<ctrl-X\>N	Move to next window.
\<ctrl-X\>P	Move to previous window.
\<ctrl-X\>Z	Enlarge current window by *argument* lines.
\<ctrl-X\>\<ctrl-N\>	
	Move text in current window down by *argument* lines.
\<ctrl-X\>\<ctrl-P\>	
	Move text in current window up by *argument* lines.
\<ctrl-X\>\<ctrl-Z\>	
	Shrink current window by *argument* lines.

Buffers

\<ctrl-X\>B	Prompt for buffer name, display that buffer in current window.
\<ctrl-X\>K	Prompt for buffer name, delete that buffer.
\<ctrl-X\>\<ctrl-B\>	
	Display change flag, size, buffer name, and file name for each buffer.
\<ctrl-X\>\<ctrl-F\>	
	Rename file linked to current buffer.
\<ctrl-X\>\<ctrl-R\>	
	Prompt for a file, and read it into current buffer, displacing its contents.

<ctrl-X><ctrl-V>

Prompt for file, display it in current window.

Saving Text and Exiting

<ctrl-X><ctrl-C>

Exit, do not save changes to text.

<ctrl-X><ctrl-S>

Save current buffer into associated file.

<ctrl-X><ctrl-W>

Prompt for file name and write current buffer into it.

<ctrl-Z> Save current buffer to associated file and exit.

Compilation Error Handling

<ctrl-X>> Move to next error.

<ctrl-X>< Move to previous error.

Search and Replace

<ctrl-R> Incremental search backward; seek a pattern as each character is typed.

<esc>R Search toward beginning of file.

<ctrl-S> Incremental search forward; seek a pattern as each character is typed.

<esc>S Search toward end of file. Waits for entire pattern before search begins.

<esc>% Search and replace. Prompt for two strings; then search for first string and replace it with second.

<esc>/ Search for next occurrence of a string entered with **<esc>S** or **<esc>R** commands; this remembers whether previous search had been forward or backward.

<esc>@ Toggle case sensitivity for searches. By default, searches are case insensitive.

Keyboard Macros

<ctrl-X>(Begin a macro definition. MicroEMACS collects everything typed until next **<ctrl-X>)** for subsequent repeated execution. **<ctrl-G>** breaks definition.

<ctrl-X>) End a macro definition.

<ctrl-X>E Execute keyboard macro.

<ctrl-X>M Bind a newly created keyboard macro to a given keystroke or set of keystrokes.

Flexible Key Bindings

<ctrl-X>R Replace one binding with another.

<ctrl-X>X Rebind prefix (meta) keys, and multiple-execution key **<ctrl-U>**.

<ctrl-X>S Prompt for a file, and write all flexible keybindings and macros into it.

<ctrl-X>L Prompt for a file, read flexible keybindings and macros from it.

<ctrl-X>I Rebind current macro to initialization macro.

Change Case of Text

<esc>C Capitalize next word.

<ctrl-X><ctrl-L>

Convert all text from current position to mark into lower case.

<esc>L Convert next word to lower case.

<ctrl-X><ctrl-U>

Convert all text from current position to mark into upper case.

<esc>U Convert next word to upper case.

White Space

<ctrl-I> Insert a tab. Default behavior is to move cursor to nearest 8's boundary; for example, if cursor is in 62nd column on screen, pressing **<ctrl-I>** moves it to column 64. When used with a positive argument, change the behavior of tab key. When used with a negative argument, change behavior of tab character. For example, **<ctrl-U>-4<ctrl-I>** says that a tab character on a file will take you to nearest 4's boundary. Thus, if you have a file with tabs in it and you use '-4', appearance of file on screen will change; but if you use '4' appearance of file on screen will not change.

<ctrl-J> Insert a new line and indent to current level. This is often used in C programs to preserve current level of indentation.

<ctrl-M> If following line is not empty, insert a new line; if empty,

move to next line.

<ctrl-O>	Open a blank line; that is, insert newline after current position.
<tab>	With argument, set tab fields at every *argument* characters. An argument of *zero* restores default of eight characters. Setting the tab to any character other than eight causes space characters to be set in your file instead of tab characters.

Send Commands to Operating System

<ctrl-C>	Suspend MicroEMACS and execute a subshell. Typing **<ctrl-D>** returns you to MicroEMACS and allows you to resume editing.
<ctrl-X>!	Prompt for a shell command and execute it.

Setting the Mark

<ctrl-@>	Set mark at current position.
<esc>.	Set mark at current position.
<ctrl><space>	Set mark at current position.

Help Window

<ctrl-X>?	Prompt for word for which information is needed.
<esc>?	Search for word over which cursor is positioned.
<esc>2	Erase help window.

Miscellaneous

<ctrl-G>	Abort a command.
<ctrl-L>	Redraw screen.
<ctrl-Q>	Insert next character into text; used to insert control characters.
<esc>Q	Quote a character by numeric value. MicroEMACS prompts for a decimal value, then inserts into the text the character whose value you type.
<ctrl-T>	Transpose the characters before and after current position.
<ctrl-U>	Specify a numeric argument, as described above.
<ctrl-U><ctrl-X><ctrl-C>	
	Abort editing and re-compilation. Use this command to abort editing and return to COHERENT when you are using **-A** option to **cc** command.
<ctrl-X>H	Use word-wrap on a region.
<ctrl-X>F	Set word wrap to *argument* column. If argument is one, set word wrap to cursor's current position.
<ctrl-X><ctrl-X>	
	Mark current position, then jump to previous setting of mark.