

UNIXTM SYSTEM V COMMAND SUMMARY

Examples of Common Commands

at 2 am Friday < file
at 2:00 a.m. Friday, execute commands in file

awk '{print \$1 + \$2}' file1
print sum of first two fields of each line

cat file1 >> file2
append file1 to end of file2

cc file.c
compile C program, executable in a.out

cd /usr/lib
change working directory to specified one

chmod g + rw file1 file2
change mode of files, adding group read and write access

chmod 600 file
change mode of file, allow only read and write by owner

cp file1 file2
make a copy of file1 named file2

cp file1 file2 /tmp
put copy of file1 and file2 into specified directory

cu -s1200 -l/dev/tty7 telno
dial up unix system at 1200 baud on line ldev/tty7

diff file1 file2
report differences between two files

f77 file.f
compile Fortran program, executable in a.out

f77 -o file file.f file.o
compile Fortran program, link with file.o, executable in file

find \$HOME -name '#*' -exec rm {} \;
remove files with names beginning with a pound sign

grep '[Pp]hone' file
print all lines in file containing Phone or phone

grep -l main *
print names of files in current directory containing main

help stuck
explain commands and messages using help

kill -9 0
send a KILL signal to processes started since login

ln file1 file2
make a link to file1 named file2

lp -m file
spool a file, send mail when printing done

lpstat -u
print status of user's line printer requests

ls -l pr -5 -t
print a five column list of files in current directory

ls -p /bin
list files in /bin directory, specify directories with l

mail molly tracey < file
send a file to specified users as mail

ED COMMANDS

Many commands are of the form *address command*. In these commands, the items in parentheses indicate default address values. Two values separated by a comma indicate an address range. *re* refers to a regular expression (see below). *nre* refers to a new (replacement) *re*.

Addressès

.	current line
\$	last line
<i>n</i>	<i>n</i> th line
' <i>x</i>	line marked as <i>x</i> with the k command
<i>lrel</i>	first line (forward) with <i>re</i>
? <i>re</i> ?	first line (backward) with <i>re</i>
[<i>addr</i>] ± [<i>n</i>]	relative to specified address (<i>addr</i> defaults to . and <i>n</i> defaults to 1)

Commands

(.) a	append; end with period alone on a line
(..) c	change; end with period alone on a line
(..) d	delete
e [<i>file</i>]	edit <i>file</i>
E [<i>file</i>]	edit <i>file</i> ; no diagnostics
f [<i>file</i>]	set current filename
(1,\$) glrelcmds	global on matching lines
(1,\$) Glrelcmds	interactive global
h	explain last ? diagnostic
H	toggle explanatory diagnostics mode
(.) i	insert; end with period alone on a line
(.. + 1) j	join
(.) kx	set mark <i>x</i> at addressed line
(..) l	list displaying special characters
(..) ma	move lines after <i>a</i>
(..) n	print with line numbers
(..) p	print lines
P	toggle . prompt mode
q	quit
Q	quit without checking for buffer changes
(\$) r [<i>file</i>]	read <i>file</i>
(..) slrelnrel	substitute
(..) slrelnrelg	global substitute
(..) slrelnreln	only match <i>n</i> 'th occurrence of <i>re</i>
(..) ta	copy lines after <i>a</i>
u	undo previous substitution
(1,\$) vlrelcmds	like g but unmatching lines
(1,\$) Vlrelcmds	like G but unmatching lines
(1,\$) w [<i>file</i>]	write
X	encrypt during r , e or x
(\$) =	print line number
! UNIXcmd	execute UNIX command
(. + 1)< nl >	print specified line

Note: If *file* (in **e**, **E** and **r**) begins with ! it is a UNIX command who's output is input to the edit buffer. In **w**, !*file* uses buffer as input to command.

Regular Expressions

<i>c</i>	the character <i>c</i>
\ <i>c</i>	character <i>c</i> (for special characters)
.	any character except newline
[<i>str</i>]	any one character in <i>str</i>
[^ <i>str</i>]	any character except <i>str</i> or newline
<i>re</i> *	0 or more occurrences of 1 character <i>re</i>
<i>re</i> { <i>m</i> , <i>n</i> }	<i>m</i> thru <i>n</i> occurrences of 1 character <i>re</i>
\(<i>re</i> \)	group <i>re</i> for later reference
\ <i>n</i>	<i>n</i> th <i>re</i> in \(\...\)
^	beginning of a line
\$	end of a line
string of <i>re</i> 's	concatenation of listed <i>re</i> 's
<null>	default <i>re</i> – last <i>re</i> encountered
&	put matched string in replacement
%	same as last replacement

UNIX™ System V

Command Summary

This booklet is designed as a reference for the experienced user and a learning tool for the newcomer. It is based on UNIX System V, Release 2.0. It includes commands accessible to the normal user; maintenance, graphics and super user only commands are not included. DOCUMENTER'S WORKBENCH commands are marked with a ▲. In cases where options differ between the DOCUMENTER'S WORKBENCH version and the standard System V Release 2.0 version, both sets of options are included. Commands specific to brands of typesetters and terminals are not included.

Command names are organized alphabetically. Several conventions are used to concisely define commands.

Each command description consists of a title line, one or more invocation lines and definitions of the fields within the command.

The title line describes the command. The invocation lines show the syntax of the command. If there are multiple ways to invoke the command, they are shown in multiple invocation lines. Within the invocation lines the following conventions are used:

- A % is used to represent the system prompt and is not typed by the user
- **Bold face** is used to represent items which must be typed exactly as shown
- *Italics* are used to represent items that are to be substituted for (such as filenames)
- Brackets [] surround items that are optional

The following special abbreviations are used to minimize redundant text:

- **stdin** represents the standard input: by default, this is the terminal keyboard
- **stdout** represents the standard output: by default, this is the terminal screen or hardcopy
- **stderr** represents the standard error: by default, this is the terminal screen or hardcopy
- *option* represents the selection of an option from the list following the command line
- *file* represents a file name
- *arg* represents an argument
- Plurals (ie, *options*, *files* or *args*) imply that multiple occurrences are permitted
- Numbered versions (eg, *file1*, *file2*) indicate a precise number of occurrences is expected

UNIX is a trademark of AT&T Bell Laboratories
PDP and VAX are registered trademarks of Digital Equipment Corporation.

MS-DOS is a trademark of Microsoft Corporation

XENIX is a trademark of Microsoft Corporation

DOCUMENTER'S WORKBENCH is a trademark of AT&T Technologies.

Dial-a-Guru is a trademark of SSC, Inc.

Copyright 1985, Specialized Systems Consultants, Inc.
P. O. Box 55549, Seattle, WA 98155
All rights reserved

ISBN 0 - 916151 - 09 - 3

300, 300s - DASI 300, 300s Terminal Handler

% 300 [options]

% 300s [options]

Options:

- + 12 set 12 pitch, 6 lpi
- dt, l, c insert null after *t* tabs, *c* chars; insert 20 nulls if line length > *l* (- d3,90,30 default)
- n *n* is 1/48 inch increments for 1/2 line feed (4 default)

4014 - Tektronix 4014 Paginator

% 4014 [options] [file]

Options:

- cn *n* column output, wait after last column
- n no erase before print
- pl page length to *l*, scale: inches, lines
- t no wait between pages

450 - DASI 450 Terminal Handler

% 450

ACCTCOM - Process Accounting

% acctcom [options] [files]

stdin read if no *files* specified and **stdin** is not a terminal or **/dev/null**, else **/usr/adm/pacct** read

Options:

- a also print average statistics
- b read backwards, most recent commands first
- C *n* only processes with CPU time > *n* seconds
- e *time* only processes existing at or before *time*
- E *time* only processes ending at or before *time*
- f print fork/exec flag and exit status
- g *group* only processes of *group* (group *id* or *name*)
- h print "hog factor," (CPU time/elapsed time)
- H *factor* only processes exceeding "hog factor"
- i print I/O counts
- l *n* only processes transferring more than *n* chars.
- k print Kcore-minutes
- l *line* only processes on **/dev/line**
- m print mean core size (default)
- n *pat* only commands matching pattern *pat*
- o *output* put records in *output*, not on **stdout**
- O *n* only processes with sys CPU > *n* seconds
- q only print average statistics
- r print "CPU factor," (user time/(sys + user time))
- s *time* only processes existing after *time*
- S *time* only processes existing at or after *time*
- t separate user and system CPU times
- u *user* only processes of *user* (*user-id*, *login-name*, # (superuser), ? (unknown user-id))
- v don't print column headings

time *hr[:min[:sec]]*

ADB - General Purpose Debugger

% **adb** [*option*] [*objfile*] [*corefile*]]

Options:

-w open both files for update

Arguments:

objfile executable file (**a.out** default)

corefile core image dump file (**core** default)

Request format:

 [*address*] [,*count*] [*command*][:]

address and *count* are expressions

address set . to *address* (**0** default)

count **1** default

Commands: format: *verb* [*modifiers*]

Verbs:

newline repeat previous command with *count* = **1**

?fmt print from *address* in *objfile* in specified *fmt*

!fmt print from *address* in *corefile* in specified *fmt*

=fmt print value of *address* in specified format

[?/]val mask

 search for *val* after and with *mask* (**1** default)

[?/]L val mask

 as above but 4 bytes instead of 2

[?/]m b1 e1 f1[?/]

 set new map values

[?/]w values write 2 byte *value* to current location

[?/]W values

 write 4 byte *value* to current location

>name assign value of dot to *name*

!cmd call new shell to execute *cmd*

Modifiers:

\$<file switch command input to *file*

\$>file append output to *file*

\$a ALGOL 68 stack backtrace

\$b print breakpoints

\$c print C backtrace (*count* specifies levels)

\$C same as **\$c**, plus variables

\$d set radix to *address*

\$e print external variable values and names

\$f print floating registers in

 single or double length

\$m print address map

\$o set integer input to octal

\$q exit from **adb**

\$r print registers and set dot to **pc**

\$s set symbol match limit to *address* (**255** default)

\$v print non-zero variables in octal

\$w set page width (**80** default)

:bcmd set breakpoint, execute *cmd* when encountered

:csignal continue process sending it *signal*

:d delete breakpoint at *address*

:k kill current subprocess

:r run *objfile* as subprocess, *count* specifies

 number of breakpoints to ignore

:ssignal like **c** but single step *count* times

ADMIN - Administer SCCS Files

% **admin** [*options*] *files*

files either begin with **s.** or are directories

names of SCCS files read from **stdin** if *files* is -

Options:

- **auser** authorize *user* to make **delta**'s
- **dflag[va]** delete specified flag (see - **f**)
- **euser** erase *user*'s **delta** authorization
- **tflag[va]** set flags with optional values:
 - b** permit branch deltas
 - chigh** set highest release (**9999** default)
 - dn** set **get**'s default delta number
 - flow** set lowest release (**1** default)
 - i** fatal error if no ID keywords
 - j** permit multiple **get**'s at once
 - l***list* lock releases not to be edited
 - m***text* *text* replaces ID keyword **%M%**
 - n** make null deltas for skipped releases
 - q***text* *text* replaces ID keyword **%Q%**
 - t***text* *text* replaces ID keyword **%Y%**
 - v[file]** **delta**'s request Modification Request numbers, *file* is a checking program
- **h** verify file integrity via SCCS checksum
- **i[file]** source for new file (**stdin** default)
 - implies - **n**, only one - **i** per **admin** command
- **m[*list*]** *list* of Modification Numbers inserted as reason for first delta
- **n** create new SCCS file, file empty if no - **i**
- **rn** set initial release delta number to *n*
- **t[file]** source of descriptive text (required with - **i** and - **n**) if no *file* descriptive text removed
- **y[*text*]** *text* is comment for initial delta
 - valid only with - **i** or - **n**
- **z** 'correct' checksum ignoring file corruption

AR - Maintain Archives and Libraries

% **ar** *key afile files*

key is one character from the set **dmpqrtx** and optionally one or more from **abcilsuv**

- c** create *afile* without message
- d** delete *files* from *afile*
- l** place temporary files in local directory (**/tmp** default)
- m[*pc posname*]** move *files* to specified place (end of file default)
- p** print *files* in *afile*
- q** quickly append *files* to end of *afile*
- r[*option*][*pc posname*]** replace *files*
 - u** only update if newer
- s** create symbol table
- t** print table of contents of *afile* (all *files* default)
- v[*option*]** verbose
 - p** precede each *file* printed with name
 - t** long listing of file information
- x** extract *files*
- afile* name of archive or library
- pc* positioning character (used with **r** or **m**)
 - a** after *posname*
 - b** before *posname*
 - i** before *posname*
- posname* name of file in the archive, used to specify where to move files

ARCV - Convert Archive File Format

% **arcv** *in out*

update archive files from PDP-11 to System 5 format

AS - Assembler (Different on PDP-11)

% **as** [*options*] *file*

Options:

- j use 'long jump' assembler (VAX only)
- m run input through **m4** preprocessor
- n turn off long/short address optimization
- o *output* name of output file (*file.o* default)
- r put assembled data in **.text** section (VAX only)
- R delete *file* after assembly
- type create **byte**, **halfword** or long displacements for undefined symbols (**l** default)
- V print assembler version number on **stderr**

ASA - Interpret Fortran ASA Carriage Controls

% **asa** [*files*]

stdin read if no *files* specified

AT - Run Command at Specified Time

% **at** *time* [*day*] [+ *n unit*]

% **at** [*option*]

Options:

- r*jobs* remove jobs scheduled by **at** or **batch**
 - l[i*jobs*] display job numbers of submitted jobs
- day* either month name followed by date, day of week, **today** or **tomorrow**, (reasonable abbreviations accepted)
- n unit* *n* **minutes**, **hours**, **days**, **weeks**, **months**, or **years**
- time* 1 - 4 digits (optional **am**, **pm**, **zulu**(GMT)) or **noon**, **midnight**, **now** or **next**

AWK - Pattern Scanning Language

% **awk** [*options*] [*prog*] [*params*] [*files*]

stdin read if - or no *files* specified

Options:

- f *pfile* use *pfile* as program
 - F*c* field separator character is *c*
- prog* program line (should be in single quotes)
- params* form **x = ..., y = ...**

BANNER - Print Banner with Large Letters

% **banner** *string*

string 10 characters maximum

BATCH - Run Command When System Load Permits

% **batch**

BASENAME - Delete Prefix and Suffix from Pathname

% **basename** *string* [*suffix*]

Arguments:

- string* path name to filter
- suffix* optional suffix to delete

BC - Unlimited Precision Arithmetic Language

% **bc** [*option*] [*files*]

stdin read after all *files*

Options:

- c compile only
- l math library (sin, cos, exp, log, arctan, Bessel)

BDIFF - Compare Big Files

% **bdiff** *file1 file2* [*options*]

stdin read if - specified for *file1* or *file2*

Options:

- n* set size of split segments to *n* (**3500** default)
- s suppress diagnostics

BFS - Big File Scanner (Read-Only **ed**)

% **bfs** [*option*] *file*

Option:

- suppress printing of *file* size

BS - Compiled/Interpreted Language

% **bs** [*file* [*args*]]

Arguments:

file read before **stdin**
args passed to program

CAL - Print Calendar

% **cal** [*month*] *year*

Arguments:

month number between 1 and 12
year number between 1 and 9999

CALENDAR - Reminder Service

% **calendar** [*option*]

Option:

- check everyone's **calendar** file and mail
 items for today and tomorrow

CANCEL - Cancel Line Printer Cmds Made with **lp**

% **cancel** [*ids*] [*printers*]

CAT - Concatenate and Print Files

% **cat** [*options*] [*files*]

stdin read if - or no *files* specified

Options:

-e display \$ before each new-line (with -v only)
-s silent about nonexistent files
-t show tabs as AI (with -v only)
-u unbuffered
-v print non-printing characters visibly
 (except tabs, new-lines and form-feeds)

CB - Beautify C Programs

% **cb** [*options*] [*file*]

stdin read if no *file* specified

Options:

-j join split lines
-l *n* split lines longer than *n*
-s use style specified in Kernighan and Ritchie

CC - C Compiler

% **cc** [*options*] *files*

unrecognized options are passed to **ld**

Options:

-B*string* substitute compiler passes
-c suppress link edit; produce .o files
-D*name*[=*def*] define *name* as *def* (1 default)
-E only preprocessor output to **stdout**
-f use floating point software
-g enable **sdb** debugger (VAX only)
-ldir search *dir* before standard ones
-o *output* name of output file (**a.out** default)
-O optimize object code produced
-p set-up object files for profiling
-P only preprocessor output to *files.i*
-S put assembler source in *files.s*
-t[*passes*] indicate which passes to substitute
-Usymb remove definition of *symb*
-W*pass*, *arg1*[*args*]

pass one or more of **p012al**

p preprocessor
0 first pass
1 second pass
2 third pass
a assembler
l link editor

CD - Change Directory

% **cd** [*directory*]

Environment variable **\$HOME** used if no *directory*

CDC - Change SCCS Delta Comments

% **cdc** [*options*] *files*

names of SCCS files read from **stdin** if *files* is -

Options:

- **m**[*list*] add Modification Request numbers, remove any preceded with an exclamation mark (!)
- **rsid** specify SCCS ID of delta to change
- **ytext** replace comment with *text*

CFLOW - Build External Reference Graph

% **cfow** [*options*] *files*

Options:

- **dn** cut off flow graph at depth *n*
- **Dname**[= *def*] define *name* as *def* (1 default)
- **i**_ include names starting with _
- **ix** include static data and external symbols (functions only default)
- **ldir** search *dir* before standard ones
- **r** produce inverted listing
- **Usymb** remove initial definition of *symb*

▲CHECKCW - Check Constant Width Text (See CW)

% **checkcw** [*options*] *files*

Options:

- **lxx** define 1 or 2 character left delimiter
- **rx** define 1 or 2 character right delimiter

▲CHECKEQ - Check **eqn** Input (See EQN)

% **checkeq** [*files*]

stdin read if no *files* specified

▲CHECKMM - Check **mm** Macro Input (See MM)

% **checkmm** [*files*]

stdin read if no *files* specified

CHGRP - Change Group ID of Files (See CHOWN)

% **chgrp** *group files*

group group name or number

CHMOD - Change Access Modes

% **chmod** *mode files*

mode can be numeric or symbolic. The symbolic case consists of the form [**agou**][+ - =][**rstwx**] where:

- a** group, other and user access permissions (default)
- g** group access permissions
- o** other access permissions
- u** user access permissions
- +** add the permission to status of *files*
- remove the permission from status of *files*
- =** set the permission of *files* to specified value
- r** read permission
- s** set owner-ID or group-ID on execution
- t** save text mode
- w** write permission
- x** execute permission

Multiple symbolic modes separated by commas

The numeric case is formed from:

- 4000** set user ID on execution
- 2000** set group ID on execution
- 1000** save text image after execution (sticky bit)
- 0X00** owner's permission, where X is OR of:
04 (read), 02 (write), 01 (execute)
- 00X0** group's permission
- 000X** other's permission

CHOWN - Change Owner of Files

% **chown** *owner files*

owner login name or decimal user-id

CMP - Compare Two Files

% **cmp** [*options*] *file1 file2*

stdin read if – specified for *file1*

Options:

- l print byte number and bytes
- s silent, return codes only

COL - Filter Reverse Line-Feeds from *stdin*

% **col** [*options*]

Options:

- b printer cannot backspace
- f forward half linefeed is OK
- p don't ignore unknown ESC sequences
- x don't convert whitespace to tabs

COMB - Combine SCCS Deltas

% **comb** [*options*] *files*

names of SCCS files read from *stdin* if *files* is –

Options:

- clist *list* of deltas to be preserved
- o access as created file instead of most recent
- psid specify oldest delta to be preserved
- s generate a shell file to produce usage report

COMM - Select or Reject Common Lines

% **comm** [–*options*] *file1 file2*

stdin read if – specified for *file1* or *file2*

Options:

- 1 suppress lines only from *file1*
- 2 suppress lines only from *file2*
- 3 suppress lines in both *file1* and *file2*

CONVERT - Format Archive and Object Files

% **convert** [*option*] *in out*

Option:

- 5 make command work just like Release 1.0
- in* with –5 option:
pre-UNIX System V Release 0 object file,
link-edited module, archive of object
files or a.out modules (VAX or 3B20 only)
without –5 option:
System V, Release 0 archive file (VAX or 3B20)
- out* with –5 option:
equivalent UNIX System V, Release 1.0 file
without –5 option:
equivalent UNIX System V, Release 2.0 file

CP - Copy Files

% **cp** *file1 file2*

make a copy of *file1* named *file2*

% **cp** *files directory*

make copies of specified *files* in *directory*

CPIO - Copy Archives

% **cpio** - **i**[**6bBcdfmrsStuv**] [*patterns*]

copy in: read **stdin**, select using *patterns*

% **cpio** - **o**[**aBcv**]

copy out: pathnames from **stdin** to **stdout**

% **cpio** - **p**[**adlmruv**] *dir*

copy out and in: read **stdin**, copy to directory *dir*

Options:

6	UNIX 6th edition format
a	reset access times of input files after copy
b	swap halfwords and bytes
B	5,120 bytes/record (for /dev/rmt? only)
c	write header info as ASCII characters
d	create directories as needed
f	copy all files not in <i>patterns</i>
l	link rather than copy whenever possible
m	retain previous file modification time
r	rename files interactively
s	swap bytes
S	swap halfwords
t	print table of contents only
u	copy disregarding age of files
v	verbose, list file names

patterns names of files to select,
specified in shell notation (***** default)

dir destination path names relative to this *dir*

CPP - C Preprocessor

% **lib/cpp** [*options*] [*input*] [*output*]

Options:

-C	pass comments (strip C-style comments default)
-Dname[= def]	define <i>name</i> as <i>def</i> (1 default)
-Idir	search <i>dir</i> before standard ones
-P	don't produce line control info for C compiler
-Usymb	remove initial definition of <i>symb</i>

input input for preprocessor (**stdin** default)

output output of preprocessor (**stdout** default)

symb reserved symbols include **ibm**, **gcos**, **os**, **tss**,
unix, **interdata**, **pdp11**, **u370**, **u3b**, **vax**, **RES**, **RT**

CRONTAB - Copy to Crontab Directory

% **crontab** [*file*]

stdin read if no *file* specified

% **crontab** [*option*]

Options:

-l	display contents of user's crontab file
-r	remove crontab from /usr/spool/cron directory

CRYPT - Encrypt/Decrypt **stdin to **stdout****

% **crypt** [*password*]

Argument:

password is the key; if not given it is prompted for

CSPLIT - Split File

% **csplit** [*options*] *file* *args*

Options:

-f prefix	name new files <i>prefix00 ... prefixn</i> (xx00 ... xxn default)
-k	leave created files if error
-s	suppress character counts

args where to split *file*, of the form:

lexprl create file from current line up to
line containing *expr*. May be
followed by $\pm n$.

%expr%

same as *lexprl*, no file creation

line create file from current line to *line*

{n} repeat previous argument *n* times

CT - Call Terminal and Start Login Process

% **ct** [*options*] *telno*

Options:

- **h** inhibit hangup
- **sspeed** set baud rate (**300** default)
- **v** send status information to **stderr**
- **wn** wait up to *n* minutes for line

telno list of phone numbers to try

CTTRACE - C Program Debugger

% **ctrace** [*options*] [*file*]

stdin read if no *file* specified

Options:

- **b** only use basic functions in trace code
- **e** print variables in floating point also
- **Dname**[= *def*] define *name* as *def* (**1** default)
- **f** *functs* trace specified functions only
- **ldir** search *dir* before standard ones
- **ln** check for looping trace output in *n* consecutively executed statements (**20** default)
- **o** print variables in octal also
- **p 's'** change trace print function to *s* (**"printf"** default)
- **P** use C preprocessor before tracing input (allows - **D**, - **l**, - **U**)
- **r** *file* use *file* rather than **runtime.c**
- **s** suppress trace output from assignments and string copies
- **tn** trace *n* variables per statement (**10** default)
- **u** print variables in unsigned also
- **Usymb** remove definition of *symb*
- **v** *functs* trace everything but specified functions
- **x** print variables in hexadecimal also

file contains C program to debug

CU - Call UNIX System

% **cu** [*options*] *telno*

for dial out

% **cu** [*options*] *dir*

for direct connect

% **cu** [*options*] *sys_name*

for dial out to system in **/usr/lib/uucp/L.sys**

Options:

- **d** print diagnostic traces
- **e** set even parity
- **h** simulate half duplex terminal
- **l** *line* line device name (default = first available)
- **m** specify direct line with modem control
- **n** prompt user for telephone number
- **o** set odd parity
- **sspeed** set baud rate (**300** default)
- **t** set-up auto-answer ASCII terminal

telno number to dial (use - for pause,
= for wait for secondary dial tone)

CUT - Cut Out Fields of File

% **cut** [- *clist*] [*files*]

% **cut** [- *flist*] [- *dchar*] [- *s*] [*files*]

stdin read if no *files* specified

Options:

- **clist** pass specified character positions
- **dc** specify field delimiter (**tab** default)
- **flist** pass specified fields
- **s** suppress lines with no delimiters

list comma separated list with optional - to indicate a range

CXREF - Produce C Cross Reference

% **cref** [*options*] *files*

Options:

- **c** print combined cross reference
- **Dname**[= *def*] define *name* as *def* (1 default)
- **ldir** search *dir* before standard ones
- **o output** name of output file
- **s** silent, don't print input filenames
- **t** format output in columns of width 80
- **Usymb** remove definition of *symb*
- **wn** width of output < *n* (80 default)

DATE - Print Current Date

% **date** [+ *format*]

output can be formatted with special characters:

- %**a** abbreviated weekday - Sun to Sat
- %**d** day - 01 to 31
- %**D** date as mm/dd/yy
- %**h** abbreviated month - Jan to Dec
- %**H** hour - 00 to 23
- %**j** julian date - 001 to 366
- %**m** month - 01 to 12
- %**M** minute - 00 to 59
- %**n** insert newline
- %**r** time in AM/PM notation
- %**S** second - 00 to 59
- %**t** insert tab character
- %**T** time as hh:mm:ss
- %**w** day of week - Sun = 0
- %**y** last 2 digits of year - 00 to 99

DC - Desk Calculator

% **dc** [*file*]

stdin read after *file*

DD - Convert and Copy File

% **dd** [*options*]

Options:

- bs** = *n* both input and output block sizes
- cbs** = *n* conversion buffer size
- conv** =
 - ascii** convert EBCDIC to ASCII
 - ebcdic** convert ASCII to EBCDIC
 - ibm** different ASCII to EBCDIC
 - lcase** map upper case to lower
 - noerror** continue on error
 - swab** swap each pair of bytes
 - sync** pad every record to value of **ibs**
 - ucase** map lower case to upper
 - multiple conversions separated by commas
- count** = *n* copy only *n* records
- ibs** = *n* input block size (512 default)
- if** = *file* input file name (**stdin** default)
- obs** = *n* output block size (512 default)
- of** = *file* output file name (**stdout** default)
- seek** = *n* seek *n* output records
- skip** = *n* skip *n* input records
- n* number of bytes with optional suffix **k**, **b** or **w** to specify multiplication by 1024, 512 or 2 or a product indicated by two numbers separated by an **x**

DELTA - Install a Change into SCCS Files

% **delta** [*options*] *files*

names of SCCS files read from **stdin** if *files* is -

Options:

- **glist** *list* of deltas to ignore
- **m[*list*]** *list* of Modification Request numbers
- **n** save edited file
- **p** print delta differences in **diff** format
- **rsid** specify SCCS ID of delta
- **s** suppress printing of new SCCS ID, etc.
- **ytext** *text* inserted as comment

▲DEROFF - Remove Formatter Constructs

% **deroff** [*options*] [*files*]

stdin read if no *files* specified

Options:

- i remove **.so** and **.nx troff** commands
- mx delete text from macro lines
 - l for **mm** macros, delete **mm** lists
 - m for **mm** macros
 - s for **ms** macros
- w build word list (1 word/line)

DF - Report Free Block Count

% **df** [*options*] [*filesystems*]

Options:

- f print only free list count
 - t print total allocated block figures
- filesystems* list of device names or mounted directory names to report (default = all mounted)

DIFF - Differential File Comparer

% **diff** [*options*] *file1 file2*

stdin read if - specified for *file1* or *file2*

Options:

- b ignore trailing blanks, all whitespace equal
- e produce **ed** script to make *file2* from *file1*
- f produce script to make *file2* from *file1* (not **ed** compatible)
- h do fast comparison (-e and -f not available)

DIFF3 - Three Way File Compare

% **diff3** [*options*] *file1 file2 file3*

Options:

- 3 **ed** script for only lines with *file3* different
- e **ed** script to add *file2* to *file3*, changes to *file1*
- x **ed** script for only lines with all 3 files different

DIFFMK - Build 'Change Mark' File for **n/troff**

% **diffmk** *old new change*

Arguments:

- old* original file
- new* updated file
- change* change mark input for **n/troff**

DIRCMP - Compare Two Directories and Print Differences

% **dircmp** [*options*] *dir1 dir2*

Options:

- d compare files with same names, make list to make files in *dir2* like those in *dir1*
- s don't print messages about identical files
- wn change output width to *n* (72 default)

DIRNAME - Delete End of Pathname (See **BASENAME**)

% **dirname** *string*

DISABLE - Disable Using Specified Printers with **lp**

% **disable** [*option*] *printers*

Options:

- c cancel jobs currently on *printers*
- r *reason* apply *reason* to *printers* up to next -r, *reason* reported by **lpstat**

DU - Summarize Disk Usage

% **du** [*options*] [*directories*]

Options:

- a generate entry for each file (directories only default)
- r complain about directories that can't be read
- s only display a grand total summary

DUMP - Dump Object File or Archive Parts

% **dump** [*options*] *files*

Options: format: *option* [*modifiers*]

- a dump archive header of each archive file
- c dump string table
- f dump *file* headers
- g dump global symbols
- h dump section headers
- l dump line number information
- o dump optional headers
- r dump relocation information
- s dump section contents
- t dump symbol table
- z *func*t dump line numbers for *func*t

Option Modifiers:

- d *n* dump sections starting at *n*
- +d *n* dump sections ending at *n*
- n *name* dump info related to *name* (with *hlrst*)
- p don't print headers
- t *index* dump indexed symbol table entry; if used with
+t indicates range
- +t *index* dump symbol table entries up to *index*
entry or specified with -t
- u underline file name
- v symbolic dump (with *actghrtz*)
- z *func*t,*n* dump line *n* for *func*t, or range
starting at *n* if +z specified
- +z *n* end of range for -z

ECHO - Echo Arguments

% **echo** [*args*]

Note: special escape conventions (place string in quotes)

- \b backspace
- \c print line without newline
- \f form feed
- \n newline
- \n octal value (8 bits), *n* must start with 0
- \r carriage return
- \t tab
- \v vertical tab
- \\ backslash

ED - Text Editor (See also: inside front cover)

% **ed** [*options*] [*file*]

Options:

- suppress counts, diagnostics, etc.
- p *prompt* specify prompt
- x work with encrypted file

EDIT - Line Text Editor

% **edit** [*option*] *files*

Options:

- +pos position file at *pos* (end of file default)
- suppress ?, counts, etc.
- l set options for editing LISP files
- r retrieve last saved version of named file after
system or editor crash (list of all saved files
default)
- R set **readonly** option
- t *tag* edit file containing *tag* and position
editor at its definition
- v equivalent to using **vi**
- x edit encrypted file

pos any editor command not containing a space

EFL - Extended Fortran Language

% **efl** [*options*] [*files*]

Options:

- # suppress comments in output program
- C include comments in output program (default)
- w suppress warning messages

name = *value*

set *name* to *value* (i.e., **system** = **unix**)

EGREP - Search File for Pattern (See **GREP**)

% **egrep** [*options*] [*expr*] [*files*]

stdin read if no *files* specified

Options:

- **b** precede line with block number
- **c** print count of matching lines only
- **e** *expr* useful if the expression starts with a -
- **f** *file* take expression from *file*
- **I** ignore case of letters in comparisons
- **l** print only names of files with matching lines
- **n** print line numbers
- **v** print non-matching lines

ENABLE - Activate Specified Printers to Use **lp**

% **enable** *printers*

ENV - Alter Environment and Execute Command

% **env** [*options*] [*command* [*args*]]

Options:

- set environment only to specified values
- name = value* set environment variable *name* to *value*
- args* passed to *command*

▲EQN - Format Mathematical Text for **troff**

% **eqn** [*options*] [*files*]

stdin read if no *files* specified

Options:

- **dx** *y* set start delimiter to *x* and end delimiter to *y*
- **fn** set to font *n*
- **pn** set sub- and superscripts in point size *n*
- **sn** set in point size *n*
- **Tdev** format for specified device

▲EQNCHAR - Specify Character Definitions for **eqn**

% **eqnchar** *files*

EX - Base Text Editor for **edit** and **vi**

% **ex** [*options*] *files*

Options:

- + *pos* position file at *pos* (end of file default)
- suppress **?**, counts, etc.
- **I** set options for editing LISP files
- **r** retrieve last saved version of named file after system or editor crash (list of all saved files default)
- **R** set **readonly** option
- **t** *tag* edit file containing *tag* and position editor at its definition
- **v** equivalent to using **vi**
- **x** edit encrypted file

pos any editor command not containing a space

EXPR - Evaluate Expression Arguments

% **expr** *args*

F77 - Fortran 77 Compiler

% **f77** [*options*] *files*

Options:

- 1 DO loops performed at least once
- 66 don't enhance Fortran 66 compatibility
- c suppress link edit; produce .o files
- C setup for run-time subscript range checking
- E use remaining chars of argument as **efl** flags
- f use floating point software
- F run **efl** or **ratfor** and produce .of files only
- g enable **sdb** debugger (VAX only)
- m apply **m4** preprocessor to **efl** and **ratfor** files
- o*output* name of output file (**a.out** default)
- onetrip DO loops performed at least once
- O invoke object code optimizer
- p setup object files for profiling
- R use remaining chars of arg as **ratfor** flags
- S produce assembly language .s files only
- u set default variable type to undefined
- U treat upper and lower case as separate
- v display diagnostics for each process
- w suppress warnings
- w66 suppress Fortran 66 compatibility warnings

FACTOR - Factor a Number and Print Prime Factors

% **factor** [*n*]

n **stdin** read if *n* not specified

FALSE - Return Unsuccessful Exit Status (See **TRUE**)

% **false**

FGREP - Search File for Pattern (See **GREP**)

% **fgrep** [*options*] [*strings*] [*files*]

stdin read if no *files* specified

Options:

- b precede line with block number
 - c print count of matching lines only
 - e *expr* useful if expression starts with a -
 - f *file* take *strings* from *file*
 - i ignore case of letters in comparisons
 - l print only names of files with matching lines
 - n print line numbers
 - v print non-matching lines
 - x print exact matches (whole line) only
- strings* list of strings separated by (escaped) newlines and enclosed in single quotes defining the list of patterns to search for

FILE - Attempt to Classify Files

% **file** [*option*] *files*

Option:

- c check magic file for format errors, no file typing
- f *arg* use *arg* as a file of file names
- m *magic* use alternate magic file (/etc/magic default)

FIND - Find Files

% **find** *path-name-list expression*

Expressions: formed from one or more primaries

Primaries:

- **atime** *n* true if file found was accessed in *n* days
- **cpio** *dev* the file found is written on *dev* in **cpio** format
- **ctime** *n* true if file found was changed in *n* days
- **depth** always true; causes entries in directory to be acted on before the directory itself
- **exec** *cmd* execute *cmd*, true if successful exit status
- **group** *name* true if file found is owned by the group *name*
- **links** *n* true if file found has *n* links
- **mtime** *n* true if file found was modified in *n* days
- **name** *file* true if *file* matches name of file found
- **newer** *file* true if file found modified later than *file*
- **ok** *cmd* like - **exec** except user prompted first
- **perm** *octal* true if permission of file found is *octal*
- **print** print name of files found, always true
- **size** *n[c]* true if file found is *n* blocks long; if *c* specified, size is in characters
- **type** *c* true if file found is:
 - b** block special file
 - c** character special file
 - d** directory
 - f** plain file
 - p** fifo or named pipe
- **user** *name* true if file found is owned by the user *name*
- \(** *expr* **\)** true if *expr* is true, used for grouping
- n* *n* means exactly *n*, + *n* means more than *n*
- *n* means less than *n*

Ways to join primaries:

- !** *expr* negate truth value of *expr*
- exp1* *exp2* true if both *exp1* and *exp2* are true
- exp1* - **o** *exp2* true if either *exp1* or *exp2* is true

▲FONT - Describe Fonts for Device-Independent troff

% **font** *files*

FSPLIT - Break efl, f77, or ratfor Files into Pieces

% **fsplit** [*options*] *files*

Options:

- **e** use **efl** input files
- **f** use **f77** input files (default)
- **r** use **ratfor** input files
- **s** remove trailing blanks, make **f77** lines ≤ 72 characters

GET - Retrieve an SCCS File Version

% **get** [*options*] *files*

names of SCCS files read from **stdin** if *files* is -

Options:

- **an** specify delta sequence number retrieved
- **b** create new branch (- **e** required)
- **cwhen** do not include deltas made after *when* (format: YY[MM[DD[HH[MM[SS]]]]])
- **e** retrieve the version for editing
- **g** suppress version retrieval
- **ilist** *list* of deltas to include
- **k** SCCS ID keywords not replaced
- **l** delta summary written to **l.file.c**
- **lp** delta summary written to **stdout**
- **m** precede each line with its related SCCS ID
- **n** precede each line with %**M**% ID keyword
- **p** retrieved version written to **stdout**
- **rsid** specify SCCS ID of version
- **s** suppress **stdout** output
- **t** access most recent delta
- **w** *str* substitute *str* for %**W**%
- **xlist** *list* of deltas to exclude

GETOPT - Parse Command Options

% **set** - - *\getopt string \$**
 string list of recognized option letters

GREEK - Setup Extended Character Set Filter

% **greek** [- *Tterm*]
 \$TERM default

Terminals:

300	DASI 300
300-12	DASI 300 in 12 pitch
300s	DASI 300s
300s-12	DASI 300s in 12 pitch
450	DASI 450
450-12	DASI 450 in 12 pitch
1620	Diablo 1620
1620-12	Diablo 1620 in 12 pitch
2621	same as hp
2640	same as hp
2645	same as hp
hp	HP 2621, 2640, and 2645
tek	Tektronix 4014

GREP - Search File for Pattern

% **grep** [*options*] *pattern* [*files*]
 stdin read if no *files* specified

Options:

- b	precede line with block number
- c	print count of matching lines only
- i	ignore case of letters in comparisons
- l	print only names of files with matching lines
- n	print line numbers
- s	suppress file error messages
- v	print non-matching lines

HELP - Explain a Message or Command

% **help** [*args*]
 args message numbers or command names
 help stuck display info on using **help**

HP - Handle Special Functions of HP2640 & 2621

% **hp** [*options*]

Options:

- e	use display enhancements mode
- m	convert more than 2 consecutive newlines to 2

HPIO - HP 2645A Terminal Tape File Archiver

% **hpio** - *i* [*options*] [- *n n*]
 copy in: extract *files* from tape

% **hpio** - *o* [*options*] *files*
 copy out: copy *files* to tape

Copy in options:

a	query before creating a file
- n n	extract <i>n</i> input files (1 default)
r	use right tape drive
t	only print table of contents

Copy out options:

c	include checksum at end of each <i>file</i>
r	use right tape drive

▲**HYPHEN** - Find Hyphenated Words and Print

% **hyphen** [*files*]
 stdin read if no *files* specified

ID - Print User and Group IDs and Names

% **id**

IPCRM - Remove Specified Inter-Process Communications

% **ipcrm** [*options*]

Options:

- **m** *mem* remove shared memory id and structure
- **M** *mkey* remove shared memory structure created with *mkey*
- **q** *msg* remove message queue id and structure
- **Q** *qkey* remove message queue structure created with *qkey*
- **s** *sem* remove semaphore id and structure
- **S** *skey* remove semaphore structure created with *skey*

IPCS - Print Inter-Process Communication Facility Status

% **ipcs** [*options*]

Options:

- **a** include all print options (-**b** -**c** -**o** -**p** -**t**)
- **b** report info about maximum allowable sizes
- **c** report creator's login and group name
- **C** *core* use *core* for reporting (**/dev/kmem** default)
- **m** report on active shared memory segments
- **N** *list* report on processes in *list* (**/unix** default)
- **o** print current message & shared memory usage
- **p** report process-ids of recently active processes
- **q** report on active message queues
- **s** report on active semaphores
- **t** report time info on recently active processes

JOIN - Form the Join of Two Relations

% **join** [*options*] *file1 file2*

stdin read if - specified for *file1*

Options:

- **an** produce line for each unpairable line in *filen*
- **e** *str* replace empty fields with string *str*
- **jn** *m* join on *m*'th field of *filen*
- **o** *list* specify output fields; each element of form: *n.m* where *n* is file and *m* is field
- **tc** use *c* as field separator (**tab** default)

KILL - Terminate or Send a Signal to Processes

% **kill** [*option*] *pids*

Option:

- **signo** decimal number of signal sent (**15** default)
- 01** hangup
- 02** interrupt
- 03** quit
- 04** illegal instruction
- 05** trace trap
- 06** IOT instruction
- 07** EMT instruction
- 08** floating point exception
- 09** kill (cannot be caught or ignored)
- 10** bus error
- 11** segmentation violation
- 12** bad system call argument
- 13** write on unread pipe
- 14** alarm clock
- 15** software termination signal
- 16** user defined signal 1
- 17** user defined signal 2
- pids* process id's to receive the signal
 (**0** implies all processes resulting from current login)

LD - Linkage Editor

% **ld** [*options*] *files*

Options:

- **e** *symb* set default entry point to value of *symb*
- **f** *fill* set default fill pattern to *fill*
- **l** *x* force load of */lib/libx.a* or */usr/lib/libx.a*
- **L** *dir* search *dir* before */lib*
- **m** print list of input/output sections on **stdout**
- **M** print message for each compiler multiply defined external definition
- **N** put data after text in output file
- **o** *output* name of output file (**a.out** default)
- **r** generate relocatable (*ld*'able) output
- **s** strip output of symbol table and relocation bits
- **t** don't warn about different sized multiply defined symbols
- **u** *symb* enter *symb* as undefined symbol
- **V** print *ld* version number on **stderr**
- **VS** *n* put version number *n* in output file header
- **x** don't preserve local symbols in table

LEX - Generate Lexical Analysis Programs

% **lex** [*options*] [*files*]

stdin read if no *files* specified

Options:

- **c** **c** actions (default)
- **n** don't print out summary
- **r** **ratfor** actions
- **t** output to **stdout** instead of **lex.yy.c**
- **v** provide summary statistics

LINE - Copy One Line from stdin to stdout

% **line**

LINT - C Program Checker

% **lint** [*options*] *files*

Options:

- **a** suppress messages about assignment of longs to non-longs
- **b** suppress messages about unreachable **breaks**
- **c** put output of 1st pass in **.ln** file
- **Dname**[= *def*] define *name* as *def* (1 default)
- **g** enable **sdb** debugger
- **h** don't apply heuristic tests
- **ldir** search *dir* before standard ones
- **lx** include library **/usr/lib/lib-lx.ln**
- **n** don't check compatibility against libraries
- **o** *lib* put input to 2nd pass in library **lib-l/lib.ln**
- **O** optimize object code produced
- **p** check portability with other C dialects
- **u** suppress messages about undefined or unused functions and external variables
- **Usymb** remove definition of *symb*
- **v** suppress messages about unused arguments
- **x** don't report unused variables declared external

LN - Make Links to Files (See CP)

% **ln** [*option*] *file1 file2*

make a link of *file1* named *file2*

% **ln** [*option*] *files directory*

make links of specified files in *directory*

Option:

- **f** force link despite target file permissions

LOGIN - Sign On to System

% **login** [*name*] [*args*]

login as user *name*, logout if no *name* specified

Argument:

env_var = *value*

set environment variables (not **PATH** or **SHELL**)

LOGNAME - Print Login Name

% **logname**

print value of **\$LOGNAME** environment variable

LORDER - Find Ordering Relation for Archive Files

% **lorder files**

LP - Send Request to Line Printer

% **lp [options] files**

Options:

- c copy rather than link files
- dptr sent request to specified printer
- m send mail after printing complete
- nn print *n* copies (1 default)
- ooption specify printer or class dependent option
- s suppress messages from **lp**
- ttitle print *title* on banner page of printout
- w write to user's terminal after printing complete

LPSTAT - Print Status of Line Printer System

% **lpstat [options]**

Options:

- a[list1] print acceptance status of request destinations
- c[classnames] print *classnames* and members
- d print default destination for **lp**
- o[list2] print status of printer requests
- p[ptrs] print status of specified printers
- r print information on the **lp** request scheduler
- s print summary system statistics
- t print all status information
- u[users] print information on *user's* requests
- v[ptrs] print list of pathnames for printers
- (default) print information on all requests made by user
- list1 list of printer and class names
- list2 list of printer names, class names, request ids
- lists comma separated

LS - List Contents of Directories

% **ls [options] [directories]**

current working directory used if no *directories* specified

Options:

- a list all entries (including ones starting with .)
- b print non-graphic characters in octal
- c use time file created in -t & -l options
- C multi-column list, sorted down each column
- d list only name (not contents) of directory
- f interpret each argument as directory
- F distinguish directories '/' executable files '.'
- g like -l but don't print owner
- i print id-number
- l long list (mode, links, owner, group, size, time of last modification)
- m comma separated list of files
- n like -l but show GID and UID as numbers
- o like -l but don't print group
- p mark directories with /
- q print non-graphic characters as ?
- r reverse sort order
- R recursively print subdirectories
- s print size in blocks
- t sort by modification time
- u use time of last access in -t & -l options
- x multi-column list, sorted across each row

M4 - Macro Preprocessor

% **m4** [*options*] [*files*]

stdin read if – or no *files* specified

Options:

- **B***n* set push-back & arg buffers to *n* (**4096** default)
- **D***name*[= *val*] define *name* to *val* (**null** default)
- **e** operate interactively
- **H***n* set hash array to prime number *n* (**199** default)
- **s** enable line sync output for C preprocessor
- **S***n* set call stack size to *n* (**100** default)
- **T***n* set size of token buffer to *n* (**512** default)
- **U***name* undefine *name*

MAIL - Send or Read Mail

% **mail** [*option*] *names*

send message from **stdin** to users specified with *names*

Send Mail Option:

- **t** include list of people **mail** sent to in message

% **mail** [*options*]

Read Mail Options:

- **e** don't print mail, return exit value 0 if mail, 1 if no mail
- **f***file* use *file* instead of **/usr/mail/user**
- **p** print all messages, no prompt for disposition
- **q** terminate after interrupt
- **r** print messages in first-in, first-out order

MAILX - Send or Read Mail

% **mailx** [*options*] [*users*]

Options:

- **d** show debugging output
- **e** check if mail is present, return 0 if mail
- **f** *file* use *file* instead of **mailbox** (**mbox** default)
- **F** put message in file named after first *user*
- **h** *n* *n* network connections made so far
- **H** only print header summary
- **i** ignore tty interrupt signals
- **n** disable reading of **/usr/lib/Mailx.rc**
- **N** don't print header summary
- **r** *address* give *address* to network delivery software
- **s** *subj* start mail with **Subject:** *subj*
- **u** *name* read *name*'s mail
- **U** convert **uucp** addresses to internet format

MAKE - Maintain Program Groups

% **make** [*options*] [*names*]

Options:

- **b** old **makefile** compatibility mode
- **d** print detailed debugging information
- **e** environment variables override **makefile**
- **f** **makefile** specify name of description file (**makefile**, **Makefile**, **s.makefile**, **s.Makefile** defaults) **makefile** named – uses **stdin**
- **i** ignore errors of invoked commands
- **k** abort current entry on error, continue with unrelated entries
- **m** print a memory map
- **n** print but don't execute commands
- **p** print macro definitions and target descriptions
- **q** successful exit status if target file is current
- **r** do not use built-in rules
- **s** do not print commands before executing them
- **t** force updating of target files by **touching** them

MAKEKEY - Make Encryption Key

% **usr/lib/makekey**

▲ MAN - Print Manual Entries

% **man** [*options*] [*section*] *titles*

Options:

- 12 produce 12 pitch output
- c invoke **col** to process output
- d search current directory instead of */usr/manV*
- s typeset in small (6"X9") format
- t typeset in default (8.5"X11") format
- T*term* format using **nroff** for terminal type *term* (450 default)
- w print only path names of entries
- y use non-compacted macros

MESG - Permit or Deny Messages via **write**

% **mesg** [*option*]

current message state printed if no *option* specified

Options:

- n deny messages
- y allow messages

MKDIR - Create Specified Directories

% **mkdir** *dirname*s

▲ MM - Print MM Format Documents

% **mm** [*options*] [*files*]

option list printed if no arguments specified

Options:

- 12 use 12 pitch print
- c invoke **col** (default except for fancy printers)
- cm use compacted version of macros
- e invoke **neqn**
- E invoke - e option of **nroff**
- mm use noncompacted version of macros
- t invoke **tbl**
- T*term* specify terminal type (**\$TERM** default)
- y use non-compacted macros

Any other options are passed to **nroff**

▲ MMT - Typeset MM Format Documents

% **mmt** [*options*] [*files*]

stdin read if - specified in *files*

Options:

- a invoke the - a option of **troff**
- D*dest* specify output destination
- e invoke **eqn**
- p invoke **pic**
- t invoke **tbl**
- T*term* specify terminal type (**\$TERM** default)
- y use non-compacted macros
- z don't direct output through post-processor

Any other arguments passed to **troff** or macro package

▲ MPTX - Macro Package for Formatting Permuted Index

% **mptx** *files*

MV - Move Files (See CP)

% **mv** [*option*] *file1 file2*

rename (or move) *file1* to *file2*

% **mv** [*option*] *files directory*

rename (or move) specified *files* to *directory*

Option:

- f force move despite target file permissions

▲ MVT - Typeset Viewgraphs (See MMT)

% **mvt** [*options*] [*files*]

stdin read if - specified in *files*

Options:

- a invoke the - a option of **troff**
- D*dest* specify output destination
- e invoke **eqn**
- p invoke **pic**
- t invoke **tbl**
- T*term* specify terminal type (**\$TERM** default)
- z don't direct output through post-processor

Any other arguments passed to **troff** or macro package

▲ **NEQN** - **nroff** Compatible Math Formatter (See **EQN**)

% **neqn** [*options*] [*files*]

stdin read if no *files* specified

Options:

- **dxy** set start delimiter to *x* and end delimiter to *y*
- **fn** set to font *n*
- **pn** set sub- and superscripts in point size *n*
- **sn** set in point size *n*
- **Tdev** format for specified device

NET - Execute Command on Remote System (DEC only)

% **net remote** [*cmd*] [*args*]

remote format: *system_name channel_number*

cmd **/bin/sh** - *i* default

NEWFORM - Change Text File Format

% **newform** [-*s*] [*options*] [*files*]

stdin read if no *files* specified

Options:

- **an** append *n* characters to end of line
- **bn** truncate *n* characters from beginning of line
- **ck** set prefix/suffix character to *k* (space default)
- **en** truncate *n* characters from end of line
- **f** write tab format before output (-8 default)
- **iformat** set tab format (-8 default)
- **ln** set line length to *n* characters (72 default)
- **oformat** replace spaces by tabs according to tab format (-8 default)
- **pn** prefix *n* characters to the beginning of line
- **s** remove characters before first tab, place up to 8 at the end of the line

NEWGRP - Login to New Group

% **newgrp** [*option*] [*group*]

user's login group used if no *group* specified

Option:

- change environment as if user logged in again

NEWS - Print News Items

% **news** [*options*] [*items*]

Options:

- **a** print all items
- **n** print names, not contents of current items
- **s** print number of current items

NICE - Run Command at Low Priority

% **nice** [*option*] *command* [*args*]

priority lowered by 10 if *option* not specified

Option:

- **n** lower scheduling priority by *n*, range 1 - 19

NL - Line Numbering Filter

% **nl** [*options*] [*file*]

stdin read if no *file* specified

Options:

- **btype** number specified lines:
Types:
 - a** all lines
 - n** no lines
 - pstring** lines containing *string*
 - t** lines with text only (default)
- **dxx** specify delimiters for start of logical page section (\: default)
- **ftype** like -b except for footer (n default)
- **htype** like -b except for header (n default)
- **in** page number increment (1 default)
- **ln** *n* blank lines treated as one (1 default)
- **nformat** line numbering format:
 - ln** left justify, zero suppressed
 - rn** right justify, zero suppressed (default)
 - rz** right justify, zero filled
- **p** don't restart numbers at logical page ends
- **sc** use *c* between number and text (tab default)
- **vn** number first page *n* (1 default)
- **wn** set size of number field (6 default)

NM - Print Symbol Table (Different on PDP-11)

% **nm** [*options*] [*files*]

Options:

- e print static and external symbols only
- f print full output, including redundant symbols
- h don't print header
- n sort external symbols by name
- o print value & size in octal (decimal default)
- T truncate symbol names; keep columns aligned
- u print undefined symbols only
- v sort external symbols by value
- V print **nm** version number on **stderr**
- x print value & size in hexadecimal

NOHUP - Run Command Ignoring Hangups

% **nohup** *command* [*args*]

▲NON-BTL - Reinstall **mm** without AT&T Bell Labs Features

% **non-btl**

▲NROFF - Format Text (See **TROFF**)

% **nroff** [*options*] [*files*]

stdin read if - or no *files* specified

Options:

- c*name* prepend compacted files
/usr/lib/macros/[u]cmp.[nt].[dt].*name*
 - e equally space words in justified lines
 - h speed output with tabs (set every 8 spaces)
 - i read **stdin** after all *files*
 - k*name* compact macros used and place in [dt].*name*
 - m*name* prepend macro file /usr/lib/tmac/tmac.*name*
 - nn number first page *n*
 - olist print only listed page numbers
 - q invoke simultaneous input/output mode of .rd
 - ran set register *a* to *n*
 - sn stop every *n* pages (1 default)
 - T*term* specify output terminal (lp, 37, 300, 300s, 382, 450, 832, 2631, 4000A, tn300, and X supported)
 - un overstrike bold characters *n* times (0 default)
 - z print only output from .tm requests
- list* comma separated, *n* - *m* means range,
- *n* means beginning to page *n*, *n* - means
from *n* to end

OD - File Dump

% **od** [*options*] [*file*] [[+]*offset* [.][*b*]]

stdin read if no *file* specified

Options:

- b byte dump in octal
 - c byte dump in ASCII characters
 - d word dump in unsigned decimal
 - o word dump in octal (default)
 - s 16 bit word dump in signed decimal
 - x word dump in hexadecimal
- offset* specify octal offset to start dumping *file*
+ required if *file* is omitted
. indicate decimal offset
b indicate offset is in 512 byte blocks

▲OSDD - Print OSDD Format Documents (See **MM**)

% **osdd** [*options*] [*files*]

option list printed if no arguments specified

Options:

- 12 use 12 pitch print
 - c invoke **col** (default except for fancy printers)
 - e invoke **neqn**
 - E invoke -e option of **nroff**
 - t invoke **tbl**
 - T*term* specify terminal type (**\$TERM** default)
 - y use non-compacted macros
- Any other options are passed to **nroff**

PACK - Compress Files

% **pack** [*option*] *files*

files are compressed to *files.z*,
original file removed

Option:

- print statistical information on **stdout**
- f force packing of *files*

PASSWD - Change Login Password

% **passwd**

PASTE - Horizontally Concatenate Files

% **paste** [*options*] *files*

stdin read if - specified in *files*

Options:

- d*list* use *list* char's as line separators (**tab** default)
- s merge subsequent lines from one file
- list* characters reused when exhausted

PCAT - Unpack and Concatenate Packed Files (See **PACK**)

% **pcat** *files*

PCC - Portable C Compiler (See **CC**)

% **pcc** [*options*] *files*

Options:

- B*string* substitute compiler passes
- c suppress link edit; produce **.o** files
- D*name*[=*def*] define *name* as *def* (**1** default)
- E only preprocessor output to **stdout**
- f use floating point software
- g enable **sdb** debugger
- ldir search *dir* before standard ones
- o *output* name of output file (**a.out** default)
- O optimize object code produced
- p set-up object files for profiling
- P only preprocessor output to *files.i*
- S put assembler source in *files.s*
- t[*passes*] indicate which passes to substitute
- U*symb* remove initial definition of *symb*
- W*pass*, *arg1*[*args*]

pass one or more of **p012a1**

PDP11 - True Exit Status if PDP-11

% **pdp11**

PG - View File by Screenful or By Line

% **pg** [*options*] [*files*]

stdin read if - or no *files* specified

Options:

- +l*patl* start at first line containing *pat*
- c clear screen before displaying new page
- e don't pause after each file
- f don't split lines
- n don't need new-line after command letters
- n window size
- +n start viewing file at line *n*
- p *prompt* change prompt; if *prompt* includes %d, insert page number in prompt
- s print messages and prompts in reverse video

h shows list of commands

▲PIC - **troff** Preprocessor for Drawing Pictures

% **pic** [*option*] [*file*]

Option:

- T*dev* specify device

PR - Print Files

% **pr** [*options*] [*files*]

stdin read if - or no *files* specified

Options:

- **a** print multi-column output across page
- **d** print double spaced
- **ecn** expand input tabs to every *n*th position using *c* as tab char (*n* = 8 default; *c* = **tab** default)
- **f** use form feed character for new page, pause before first page if **stdout** is to terminal
- **h head** use *head* as heading line (file name default)
- **icn** convert whitespace to tabs every *n*th position *c* as tab char (*n* = 8 default; *c* = **tab** default)
- **ln** set page length to *n* lines (66 default, 11 min)
- **m** merge and print all *files*, one per column
- + **n** begin printing at page *n* (1 default)
- **n** produce *n* column output (1 default)
- **ncn** number lines with *n*-wide numbers followed by *c* (*n* = 5 default; *c* = **tab** default)
- **on** set line offset to *n* (0 default)
- **p** pause between pages if output is to a terminal
- **r** no error messages if *files* cannot be opened
- **sc** set column separator to *c* (**tab** default)
- **t** don't print page heading or trailing lines
- **wn** set line width to *n* (72 default for equal width multi-column output, no limit otherwise)

PROF - Display Profile Data

% **prof** [*options*] [*file*]

a.out used if no *file* specified

Options:

- **a** sort by increasing symbol address
- **c** sort by decreasing number of calls
- **g** include static functions
- **h** don't print report heading
- **m profile** name of profile data source (**mon.out** default)
- **n** sort lexically by symbol name
- **o** print symbol address in octal with symbol name
- **s** print summary on **stderr**
- **t** sort by decreasing percentage of total time (default)
- **x** print symbol address in hex. with symbol name
- **z** include symbols even if zero calls and time

PRS - Print Parts of SCCS Files

% **prs** [*options*] *files*

names of SCCS files read from **stdin** if *files* is -

Options:

- **a** include removed deltas
- **c[date]** cutoff date and time in format: YY[MM[DD[HH[MM[SS]]]]]
- **dtext** output specification (includes data keys)
- **e** include deltas at *sid* and earlier (see -r)
- **l** include deltas at *sid* and later (see -r)
- **rsid** specify SCCS ID of version

PS - Report Process Status

% **ps** [*options*]

Options:

- **a** print all processes except group leaders and non-terminal associated
 - **c file** use *file* for core image (*/dev/mem* default)
 - **d** print all processes except group leaders
 - **e** print all processes
 - **f** print full listing
 - **g list** list only processes whose leaders are in *list*
 - **l** long listing (more info than - f)
 - **n list** use *list* for namelist (*/unix* default)
 - **p list** list only processes whose ids are in *list*
 - **s dev** use *dev* for swap device (*/dev/swap* default)
 - **t list** list only processes of terminals in *list*
 - **u list** list only processes with user-IDs in *list*
- list* comma or blank separated list
with optional enclosing double quotes

▲PTX - Permuted Index

% **ptx** [*options*] [*input* [*output*]]

stdin and **stdout** used if *input* or *output* not specified

Options:

- **b file** use characters in *file* to separate words (**tab**, **\n** and **space** are default separators)
- **f** sort upper and lower case together
- **g n** set gap between output parts to *n* characters (3 default)
- **i file** don't use words in *file* as keywords; if neither - **i** or - **o** are used, */usr/lib/eign* is used as *file*
- **o file** use only words in *file* as keys
- **r** treat leading characters as reference ID
- **t** prepare output for **troff** (**nroff** default)
- **w n** output line length equals *n* (72 default for **nroff**, 100 for **troff**)

PWD - Print Working Directory Name

% **pwd**

RATFOR - Rational FORTRAN Translator

% **ratfor** [*options*] [*files*]

Options:

- **6c** make continuation char *c* and put in column 6 (& in column 1 default)
- **C** copy comments to output
- **h** make quoted strings into **27H** constructs

RED - Restricted Version of **ed** Text Editor

% **red** [*options*] [*file*]

Options:

- suppress counts, diagnostics, etc.
- **p prompt** specify prompt
- **x** work with encrypted file

REGCMP - Compile Regular Expression

% **regcmp** [*option*] *files*

compile regular expression in *file* into *file.i*

Option:

- place output in *file.c* instead of *file.i*

RJESTAT - Report RJE Status, Simulate IBM Remote Console

% **rjestat** [*ibms*] [- *sibm*] [- *cibm cmd*] [- *jibm job*]

Arguments:

- *cibm cmd* interpret *cmd* as if in remote console mode
- *jibm job* report status of *job* on *ibm*
- *sibm* begin interactive status console to *ibm* after processing arguments

ibm host name (from */usr/rjellines*) to report on

RM - Remove Files

% **rm** [*options*] *files*

Options:

- **f** force removal of files without write permission
- **i** ask for confirmation before each delete
- **r** recursively delete directories

RMDEL - Remove an SCCS Delta Version

% **rmdel** -rsid files

names of SCCS files read from **stdin** if files is -

Argument:

-rsid specify SCCS ID of version to be removed

RMDIR - Remove Empty Directories (See **RM**)

% **rmdir** directories

RSH - Restricted Shell (See **SH**)

% **rsh** [options] [args]

New Options:

- a mark modified export variables
- c cmd execute cmd (default reads commands from file named in first entry of args)
- e if non-interactive, exit if a command fails
- f disable wildcarding
- h locate functions on definition instead of at execution
- i set interactive mode
- k all keyword arguments placed in environment
- n read commands without executing them
- r set restricted mode
- s read commands from **stdin**
- t read and execute one command, then exit
- u set error upon substituting an unset variable
- v print input lines as read
- x print commands, as executed, with arguments

SACT - Print SCCS Files with Versions for Editing

% **sact** files

names of SCCS files read from **stdin** if files is -

SAR - Report System Activity

% **sar** [data_options] [-o file] t [n]

sample current system activity counters

% **sar** [data_options] [-s start] [-e end] [-i sec] [-f file]

extract system information from previously stored file

Data Options:

- a file access routines
- A all data activity
- b buffer activity
- c system calls
- d block device activity
- m message and semaphore activity
- q queue activity
- u cpu utilization (default)
- v text, process, inode and file table status
- w system swapping and switching
- y tty device activity

Options:

- e end ending time of report (hh[:mm[:ss]])
- f file extract information from file
(/usr/adm/sa/sadd default, dd is current day)
- i n sample report every n seconds
(all intervals in data file default)
- n sample system for n intervals (1 default)
- o file save samples in binary format in file
- s start starting time of report (hh[:mm[:ss]])
- t each system sample t seconds long

SCC - C Compiler to Generate Stand-Alone Programs

% **scc** [*lib*] [*options*] [*files*]

lib can be one of the following:

- + don't load configuration library
- + **A** load RP04/05/06 and TU16 library (default)
- + **B** load RK11/05, RP11/03 and TM11/TU16 library

Options:

- **Bstring** substitute compiler passes
- **c** suppress link edit; produce .o files
- **Dname**[= *def*]
define *name* as *def* (1 default)
- **E** only preprocessor output to **stdout**
- **f** use floating point software
- **g** enable **sdb** debugger (VAX only)
- **ldir** search *dir* before standard ones
- **o output** name of output file (**a.out** default)
- **O** optimize object code produced
- **p** setup object files for profiling
- **P** only preprocessor output to *files.i*
- **S** only assembler source to *files.s*
- **t[passes]** indicate which passes to substitute
- **U symb** remove initial definition of *symb*
- **Wpass, arg1[args]**
 - **dn** offset *n* bytes for externals

pass one or more of **p102a1**

SCCS - Source Code Control System

See **admin**, **cdc**, **comb**, **delta**, **get**, **prs**, **rm del**,
sact, **sccsdiff**, **unget**, **val**, and **what**

SCCSDIFF - Print Differences Between Two SCCS Versions

% **sccsdiff** - *rsid1* - *rsid2* [*options*] *files*

Options:

- **p** format output using **pr**
- **sn** file segment size of *n*

SDB - Symbolic Debugger for C & F77

% **sdb** [*options*] [*objfile*] [*corefile*] [*directories*]]]

Options:

- **w** overwriting locations in *objfile* permitted
- **W** suppress warnings

Arguments:

- objfile* executable file compiled with - **g** option
(**a.out** default)
- corefile* core image dump file; - causes all core image
files to be ignored (**core** default)
- directories* default location for source files

Addresses:

proc:variable[,num]

- proc* current procedure default
- num* optional occurrence on stack, most current 1st
- [m;n]* range of subscripts (* indicates all)
- file:n* (or *proc:n*) line within a file

Commands (for data):

- n?lf* show value at line *n* in format *f* (l default)
- n = [lf]* show address of line in format *f* (lx default)
- t** show stack trace
- T** show top line of stack trace
- var?lf* show contents of *var* in format *f* (l default)
- var = [lf]* show address of *var* in format *f* (lx default)
- var[nlf]* show contents of *var*, length *l* in format *f*;
show region of memory of *n* units of length *l*
- var:value* set *var* to *value*
- x** show registers and current instruction
- X** show current machine instruction

(continued)

Formats:

<i>l</i>	length specifier (use with format cduox only); byte, half word, long word
<i>f</i>	format: character, decimal, unsigned, octal, hexadecimal, floating point (32 bit), g (floating point, 64 bit), string pointer, address of var, pointer to procedures, machine language instruction with numeric and symbolic addresses, machine instruction with numeric addresses only

Commands (for source files):

<i>lrel</i>	search forward for line containing <i>re</i>
<i>?re?</i>	search backward for line containing <i>re</i>
e <i>dir</i>	look in <i>dir</i> for source files (current default)
e <i>dir file</i>	look in <i>dir</i> for source files (current default)
e <i>name</i>	set current file to file containing file <i>name</i>
e <i>proc</i>	set current file to file containing <i>proc</i>
<i>inc +</i>	advance current line by <i>inc</i> lines, print line
<i>inc -</i>	go back <i>inc</i> lines, print line
<i>n</i>	set current line to <i>n</i> , print line
p	print current line
w	show 10 lines around the current line
z	print current line plus next 9 lines

Commands (for execution of source program):

<i>addr:m inc</i>	single step until specified location is modified
B	show currently active breakpoints
D	delete all breakpoints
I	single step one instruction; reactivate signal
I	single step one instruction; ignore signal
<i>[inc] r [args]</i>	if no <i>args</i> , reuse previous <i>args</i> ; otherwise run program with <i>args</i> ; if <i>args</i> begins with > redirect stdin , if <, stdout ; <i>inc</i> specifies number of breakpoints to ignore
<i>[inc] R</i>	run program with no <i>args</i> ; <i>inc</i> specifies number of breakpoints to ignore
k	stop program that is being debugged
l	display last executed line
<i>[l] v</i>	toggle verbose mode (with sSm); if no <i>l</i> , show changed source file and/or subroutine names; <i>l</i> if ≥ 1 , show source line before executing if ≥ 2 , also print assembler statements
<i>n a</i>	announce: if <i>n</i> is of form <i>proc</i> ., print top level of stack, otherwise, print last executed line
<i>n b [cmds]</i>	set breakpoint at line <i>n</i> ; if no <i>cmds</i> , stop before breakpoint; otherwise execute <i>cmds</i> when breakpoint is reached <i>cmds</i> semicolon separated list
<i>n c [inc]</i>	continue after breakpoint or interrupt; ignore signal; ignore <i>inc</i> - 1 breakpoints
<i>n C [inc]</i>	continue after breakpoint or interrupt; reactivate signal; ignore <i>inc</i> - 1 breakpoints
<i>n [d]</i>	delete breakpoint at line <i>n</i> ; if no <i>d</i> , interactive delete
<i>n g [inc]</i>	continue after breakpoint, resume at line <i>n</i> ; <i>inc</i> specifies number of breakpoints to ignore
<i>proc(args)</i>	run named procedure with <i>args</i>
<i>proc(args) lm</i>	run named procedure, show returned value in format <i>m</i> (d is default format)
s [inc]	single step through <i>inc</i> lines; if no <i>inc</i> , run one line of program
S inc	like s , except step through procedure calls
<i>var \$m inc</i>	single step until specified location is modified

(continued)

Miscellaneous Commands:

!*command* pass *command* to **sh**
"*string* print *string*
< *file* read commands from *file*
AD show next 10 lines: instructions, source or data
M display address maps
M **[?/]** **[*]** **b e f**
enter new values for address map:
 ? text map
 / data map
 ***** modify second segment
 b beginning
 e end
 f file offset
new-line display next source line or memory location
q exit debugger

Debugging Commands:

Q list procedures and files being debugged
V display version number
Y toggle debugger

SDIFF - Side-By-Side Difference

% **sdiff** [*options*] *file1 file2*

Options:

-l print only identical lines on left side
 -o file merge *file1* and *file2* to *file*, identical lines are passed directly, else user prompted:
 e edit an empty file
 e b edit both left and right columns
 e l edit left column
 e r edit right column
 l append left column
 q exit
 r append right column
 s suppress printing of identical lines
 v enable printing of identical lines
 -s suppress printing identical lines
 -w n set width of output line to *n* (130 default)

SED - Stream Editor

% **sed** [*options*] [*files*]

stdin read if no *files* specified

Options:

-e script editor commands in *script* executed
 -f file editor commands read from *file*
 -n suppress unrequested output

SEND - Send Files to RJE System

% **send** [*arguments*]

Arguments:

.	close current source
-	open stdin as new source
+	open terminal as new source
\$args	execute contiguous <i>args</i> ; open stdout of shell as new source
!cmd	execute <i>cmd</i> ; open stdout of <i>cmd</i> as new source
@dir	change send directory to <i>dir</i>
<i>file</i>	use <i>file</i> as new source of input
- <i>flags</i>	set <i>flag</i> (see below)
+ <i>flags</i>	reset <i>flags</i> (see below)
= <i>flags</i>	set <i>flags</i> to state at previous level (see below)
<i>host</i>	job submitted to computer <i>host</i>
? <i>key</i>	prompt if no definition for <i>key</i>
? <i>key</i> = <i>Λxx</i>	define undefined <i>key</i> as 2 digit hex code
? <i>key</i> = <i>str</i>	define undefined <i>key</i> as string <i>str</i>
<i>key</i> = <i>string</i>	set <i>key</i> to <i>string</i>
<i>key</i> = <i>Λxx</i>	define <i>key</i> as two digit hex code
= <i>key</i>	prompt for definition of <i>key</i>
: <i>message</i>	print <i>message</i> on terminal
-: <i>prompt</i>	open stdin ; if terminal, print <i>prompt</i>
+: <i>prompt</i>	open terminal and print <i>prompt</i>
: <i>spec</i> :	set default format specification for included sources
~ <i>str</i>	ignore string <i>str</i>

Flags:

- a	protect sources from key substitution
- c	print control lines on stderr
- d	append active definitions to included sources
- f	treat lower case and upper case separately
- g	list text before converting to card images
- h	print listing with tabs
- i	treat control lines as text; don't interpret
- k	erase key definitions at current level, ignore ones from previous level
- l	print card image to stdout ; EBCDIC converted to ASCII
- m	repeat prompt when input from lower level
- p	prompt with • for terminal input
- q	don't output card images
- r	pack included sources 80 chars/card until EOF
- s	substitute keys before control lines detected
- t	notify when input sources opened
- x	transmit jobs in submission order
- y	suppress error diagnostics

SH - Bourne Shell (See also: inside back cover)

% **sh** [*options*] [*args*]

Options:

- a	mark modified export variables
- c cmd	execute <i>cmd</i> (default reads commands from file named in first entry of <i>args</i>)
- e	if non-interactive, exit if a command fails
- f	disable wildcarding
- h	locate functions on definition instead of at execution
- i	set interactive mode
- k	all keyword arguments placed in environment
- n	read commands without executing them
- r	set restricted mode (See rsh)
- s	read commands from stdin
- t	read and execute one command, then exit
- u	set error upon substituting an unset variable
- v	print input lines as read
- x	print commands, as executed, with arguments

SHL - Manage Shell Layers

% **shl**

help or **?** shows summary of commands

SIZE - Size of Object File

% **size** [*options*] [*files*]

a out read if no *files* specified

Options:

- **o** print number in octal (decimal default)
- **V** print **size** version number on **stderr**
- **x** print number in hexadecimal

SLEEP - Suspend Execution for Specified Number of Seconds

% **sleep** *seconds*

SNO - Snobol Interpreter

% **sno** [*files*]

stdin read after *files*

SORT - Sort/Merge Files

% **sort** [*options*] [*files*]

stdin read if - or no *files* specified

Options:

- **b** ignore leading tabs and spaces
- **c** check that input is in sorted order
- **d** dictionary order (use only letters, digits, tabs, and spaces)
- **f** sort upper case and lower case letters together
- **i** ignore non-printing characters in comparisons
- **m** merge already sorted *files*
- **M** sort as if field contains months
- **n** numeric sort (implies - **b**)
- **o output** place sorted results in *output*
- **r** reverse sort; descending order
- **tc** set field separator to **c** (**tab** default)
- **u** output only one occurrence of duplicate lines
- **ymem** specify amount of main memory in kilobytes to begin sort (- **y0** minimum, - **y** maximum)
- **zsize** specify size of longest line read
- + *pos1* [- *pos2*]
 - sort only from *pos1* to *pos2*
 - if *pos2* not specified, key includes up to the end of line
 - pos1* and *pos2* of the form: *m*[*.n*](*bdfinr*)
 - m* *m* fields from start of line skipped (0 default)
 - n* *n* characters from start of field skipped (0 default)
 - bdfinr* option applies only to specified key

SPELL - Find Spelling Errors

% **spell** [*options*] [*files*]

stdin read if no *files* specified

Options:

- + *local* remove all words found in *local* from output
- **b** check British spelling
- **l** run **spell** on all included files
- **v** print words not literally in list and derivations
- **x** print stems for each word

SPLIT - Break File into Pieces

% **split** [*option*] [*file* [*name*]]

stdin read if - or no *file* specified

Options:

- *n* set size of split files to *n* lines
- name* output *nameaa*, *nameab*, ... (*xaa* default)

STRIP - Remove Symbol Table and Relocation Bits

% **strip** [*options*] *files*

Options:

- l strip line number info only
- r reset relocation indexes in symbol table
- V print **strip** version number on **stderr**
- x don't strip external or static symbol info

STTY - Set Terminal Options

% **stty** [-a] [-g] [*options*]

Options:

- a print all option settings
- g print settings in **stty** argument format
- 0 hang up phone line
- [-]brkint [do not] send **INTR** signal on input break
- bsn set output delay after backspace (0 or 1)
- [-]clocal [enable]/disable modem control
- cooked disables raw input and output (same as -raw)
- [-]cread [disable]/enable receiver
- crn set output delay after carriage return (0 to 3)
- csn set character size to *n* bits (5 to 8)
- [-]cstopb set [one]/two stop bits per character
- [-]echo [do not] echo all input characters
- [-]echoe [do not] echo **ERASE** for CRTs
- [-]echok [do not] echo a newline after **KILL**
- [-]echonl [do not] echo newlines
- ek reset **ERASE** to # and **KILL** to @
- eof *c* set end of file character to *c*
- eol *c* set end of line character to *c*
- erase *c* set character **ERASE** character to *c*
- [-]evenp same as [-]parenb and cs[8]/7
- ffn set output delay after form-feed (0 or 1)
- [-]hup [do not]/do hang up on last close
- [-]hupcl same as **hup**
- [-]icanon [disable]/enable checking for **ERASE** and **KILL**
- [-]ignbrk [do not] ignore break on input
- [-]ignpar [do not] ignore parity errors
- [-]inlcr [do not] map input newline to carriage return
- [-]inpck [disable]/enable input parity check
- intr *c* set **INTR** (interrupt) character to *c*
- [-]isig [disable]/enable checking for **INTR** and **QUIT**
- [-]istrip [do not] strip 8th bit of input characters
- [-]iuclc [do not] map input upper case to lower case
- [-]ixany allow [XON]/any character to restart **XOFF**
- [-]ixoff [disable]/enable **XON/XOFF** during input
- [-]ixon [disable]/enable **XON/XOFF** protocol
- kill *c* set line **KILL** character to *c*
- [-]lcase same as [-]xcase, [-]iuclic, and [-]olcuc
- [-]LCASE same as [-]lcase
- line *n* set line discipline to *n* (0 to 127 allowed)
- [-]loblk [do not] block output from non-current layer
- min *c* set **MIN** value to *c* (used with -icanon)
- n* set terminal baud rate to *n*
- [-]nl same as [icml]/- icml and [onlcr]/- onlcr
[and - inlcr, - igncr, - ocml, and - onlret]
- nl *n* set output delay after newline (0 or 1)
- [-]noflsh [do not] flush after **INTR** or **QUIT**
- [-]ocrnl [do not] map output carriage return to newline
- [-]oddp same as [-]parenb, [-]parodd, and cs[8]/7
- [-]ofdel set fill character to [NULL]/DEL
- [-]ofill delay output with [timing]/fill characters
- [-]olcuc [do not] map output lower case to upper case
- [-]onlcr [do not] map output newline to carriage return
- [-]onlret terminal does[not] carriage return after newline
- [-]onocr [do]/don't output carriage return at column 0
- [-]opost [do not] post-process output
- [-]parenb [disable]/enable parity detection & generation
- [-]parity same as [-]parenb and cs[8]/7

(continued)

[-]parmrk [do not] mark parity errors
 [-]parodd select [even]/odd parity
 quit c set **QUIT** character to c
 [-]raw [disable]/enable raw input and output
 sane reset all modes to 'sane' values
 [-]stappl use application/[line] mode on sync. line
 [-]stflush enable/[disable] **flush** after **write**
 [-]stwrap disable/[enable] line shortening
 switch c exit to **shl** command from layer (**ΛZ** default)
 tabn set output delay after horizontal tab (0 to 3)
 [-]tabs [expand to spaces]/preserve output tabs
 term set all modes for the specified terminal, *term*
 (tty33, tty37, vt05, tn300, ti700, and tek allowed)
 time c set **TIME** value to c (used with - icanon)
 vtn set output delay after vertical tab (0 or 1)
 [-]xcase [do not] change case on local output

SU - Become Another User

% **su** [*option*] [*name* [*args*]]

Option:

- change environment as if user logged in

SUM - Compute File Checksum

% **sum** [*option*] *file*

Option:

-r use alternate checksum algorithm

SYNC - Write Unwritten Info in Memory to Disk

% **sync**

TABS - Set Terminal Tabs

% **tabs** [*taboption*] [+ *mn*] [- *Ttype*]

Taboption: set tabs at columns, only 1 option

- *file* first line of *file* read for tab specifier
 -8 standard tabs, every 8 columns (default)
 -a 1,10,16,36,72 (Assembler, IBM S/370)
 -a2 1,10,16,40,72 (alternate Assembler, IBM S/370)
 -c 1,8,12,16,20,55 (normal COBOL)
 -c2 1,6,10,14,49 (compact COBOL)
 -c3 1,6,10,14,18,22,26,30,34,38,42,46,50,54,
 58,62,67 (COBOL)
 -f 1,7,11,15,19,23 (FORTRAN)
 -n 1•*n*, 2•*n*, ... (-8 default)
n1,n2,... arbitrary ascending values (up to 40 tab stops)
 if a number is preceded by + it is added
 -p 1,5,9,13,17,21,25,29,33,37,41,45,49,53,57,61 (PL/I)
 -s 1,10,55 (SNOBOL)
 -u 1,12,20,44 (UNIVAC 1100 Assembler)
 + *mn* left margin, added to tab stops (10 default)
 - *Ttype* terminal type (**\$TERM** default)

TAIL - Output Last Part of File

% **tail** [*options*] [*file*]

stdin read if *file* not specified

Options:

-f follow growth of *file* (don't stop at end of file)
 + *n*[*bcl*] begin *n* units from beginning of *file*,
 may be blocks, characters, or lines (default)
 - *n*[*bcl*] begin *n* units before end of *file* (10 default)

TAR - Tape File Archiver

% **tar** [*key*] [*files*]

stdin read if no *files* specified

Key: Format: *letter* [*modifier*]

Key Letters:

c	create new tape and record <i>files</i> , implies r
r	record <i>files</i> onto end of tape
t	tell when <i>files</i> found, all entries if no <i>files</i>
u	update tapes by adding <i>files</i> if not on tape or if modified since being last written to tape
x	extract <i>files</i> , entire tape if no <i>files</i>

Key Modifiers:

#density	# is tape drive number (0...7), (0 default)
h	high (6250 bpi)
l	low (800 bpi)
m	medium (1600 bpi) (default)
b n	<i>n</i> is blocking factor (1 default, 20 max)
f arch	<i>arch</i> is the file to be used for input/output to archives (if - then stdin read, <i>/dev/mt?</i> default)
l	complain if all file links not found
m	update file modification times
o	set user and group id of extracted files to user running tar
v	verbose mode
w	wait for confirmation after reporting file name (y causes action to be performed)

▲TBL - Format Tables (n/troff)

% **tbl** [*option*] [*files*]

stdin read if no *files* specified

Option:

- **TX** force use of only full vertical line movements

TEE - Copy **stdin** to **stdout** and Files

% **tee** [*options*] [*files*]

Options:

- **a** append to *files* instead of overwriting
- **i** ignore interrupts

TEST - Condition Evaluation

% **test** *expression*

% [*expression*]

Expressions:

- b file	true if <i>file</i> exists and is a block special file
- c file	true if <i>file</i> exists and is a character special file
- d file	true if <i>file</i> exists and is a directory
- f file	true if <i>file</i> exists and is a regular file
- g file	true if <i>file</i> exists and has set-GID bit set
- k file	true if <i>file</i> exists and has sticky bit set
- n string	true if <i>string</i> is of non-zero length
<i>n1</i> - eq <i>n2</i>	true if integers <i>n1</i> and <i>n2</i> equal
<i>n1</i> - ge <i>n2</i>	true if integer <i>n1</i> ≥ and <i>n2</i>
<i>n1</i> - gt <i>n2</i>	true if integer <i>n1</i> > <i>n2</i>
<i>n1</i> - le <i>n2</i>	true if integer <i>n1</i> ≤ <i>n2</i>
<i>n1</i> - lt <i>n2</i>	true if integer <i>n1</i> < <i>n2</i>
<i>n1</i> - ne <i>n2</i>	true if integers <i>n1</i> and <i>n2</i> unequal
- p file	true if <i>file</i> exists and is a named pipe
- r file	true if <i>file</i> exists and is readable
- s file	true if <i>file</i> exists and has a non-zero size
<i>string</i>	true if <i>string</i> is not the null string
<i>s1</i> = <i>s2</i>	true if strings <i>s1</i> and <i>s2</i> are the same
<i>s1</i> != <i>s2</i>	true if strings <i>s1</i> and <i>s2</i> are not the same
- t [fd]	true if descriptor <i>fd</i> associated with terminal
- u file	true if <i>file</i> exists and has set-UID bit set
- w file	true if <i>file</i> exists and is writable
- x file	true if <i>file</i> exists and is executable
- z string	true if <i>string</i> has zero length

Expressions may be joined by

!	logical negation
- a	logical and
- o	logical or
\(expr\)	grouping parentheses (escaped from shell)

TIME - Print a Command's Elapsed, System and User Times
% **time** *cmd*

TIMEX - Print a Command's Time and System Activity
% **timex** [*options*] *cmd*

Options:

- o report number of blocks read and written and total characters transferred
- p[*opt*] report process activity for *cmd* and its children
 - f print fork/exec flag & exit status
 - h print CPU time/elapsed time
 - k print Kcore-minutes
 - m print mean core size
 - r print user time/(sys + user time)
 - t separate user and system CPU times
- s report all system activity during *cmd* execution

TOUCH - Update File Access/Modification Times
% **touch** [*options*] *files*

Options:

- a update only access time
 - c do not create non-existent *files*
 - m update only modification time
- MMDDhhmm*[*yy*]
new times (current time default)

TPUT - Find Out Terminal-Dependent Capabilities
% **tput** [*option*] *cap*

Option:

- T*type* specify terminal type (**\$TERM** default)
- cap* name of capability in *terminfo* data base

TR - Translate Characters
% **tr** [*options*] [*string1*] [*string2*]

Options:

- c complement *string1* with 001 - 377 (octal)
- d delete characters in *string1* from input
- s squeeze repeated output characters in *string2*

Strings may include:

- [*a - z*] short form for range of characters from *a* to *z*
- [*a • n*] short form for *n* repetitions of character *a*

▲TROFF - Typeset Text
% **troff** [*options*] [*files*]

stdin read if - or no *files* specified

Options:

- a output ASCII approximation to **stdout**
 - b report whether phototypesetter is available
 - f don't feed out paper and stop phototypesetter
 - F*dir* take font info from *dir/devname*
 - l read **stdin** after all *files*
 - m*name* prepend macro file */usr/lib/tmac/tmac.name*
 - nn number first page *n*
 - olist print only listed page numbers
 - q invoke simultaneous input/output mode of *.rd*
 - ran set register *a* to *n*
 - sn stop every *n* pages (1 default)
 - t output to **stdout** instead of phototypesetter
 - T*dest* specify destination for output
 - w wait until phototypesetter is not busy
 - z print only output from *.tm* requests
- //st* comma separated, *n - m* means range,

TRUE - Return Successful Exit Status
% **true**

TSORT - Topological Sort
% **tsort** [*file*]
stdin read if no *file* specified

TTY - Display Terminal's Name

% **tty** [*option*]

Options:

- l print synchronous line number if connected
- s silent mode, no output: successful exit status if **stdin** is a terminal

U3B - Return True Exit Status If Using a 3B20

% **u3b**

U3B5 - Return True Exit Status if Using a 3B5

% **u3b5**

UMASK - Set File Creation Mask

% **umask** [*option*]

if *option* not specified, current mask printed

Option:

- ugo* 3 digit octal code specifying denied file access permissions. Each of the *ugo* digits formed of *read* (04), *write* (02), & *execute* (01) permissions for the classifications of *user*, *group*, & *others*.

UNAME - Print System Name

% **uname** [*options*]

Options:

- a print all information
- m print hardware name
- n print node name
- r print operating system release
- s print system name (default)
- v print version number of operating system

UNGET - Void SCCS File Gets

% **unget** [*options*] *files*

names of SCCS files read from **stdin** if *files* is -

Options:

- n do not remove file retrieved with **get**
- rsid specify SCCS ID of version to void
- s suppress output of SCCS ID on **stdout**

UNIQ - Report Repeated Lines

% **uniq** [*options*] [*input* [*output*]]

stdin read if *input* and *output* not specified

stdout written if *output* not specified

Options:

- c output unique lines, count repeated ones
- d only one copy of repeated lines output
- n skip *n* fields from start of line
- +n skip *n* characters from start of field
- u only unique lines in *input* output (default also outputs one occurrence of repeated lines)

UNITS - Interactive Measurement Units Conversion

% **units**

UNPACK - Unpack Compressed File (See **PACK**)

% **unpack** *files.z*

unpack from *files.z* to *files*

UUCP - UNIX to UNIX Copy

% **uucp** [*options*] *files dest*

Options:

- c use *files* directly when copying (default)
 - C copy *files* to spool directory before transmit
 - d make all required directories (default)
 - esys execute the **uucp** command on remote system
 - f do not make non-existent directories
 - j reverse action of **JOBNO** envir. variable
 - mfile report transfer status to *file*, mail when complete if no *file* specified
 - nuser notify *user* on remote system when *file* sent
 - r queue *files* but don't initiate file transfer
- dest* destination consists of [*sys_name*!]*pathname*

UULOG - **uucp** Log Maintainer (See **UUCP**)

% **uulog** [*options*]

Options:

- **ssys** print information on work with system **sys**
- **uuser** print information on work done for **user**

UUNAME - List **uucp** Names of Systems (See **UUCP**)

% **uuname** [*option*]

Option:

- **l** print local system name
- **v** print additional info on systems

UUPICK - Accept/Reject **uuto** Files (See **UUTO**)

% **uupick** [*option*]

Option:

- **ssys** only search **PUBDIR** for files from system **sys**

UUSTAT - **uucp** Status and Job Control

% **uustat** [*options*]

Options:

- **cn** remove status entries older than *n* hours
- **jn** status of job request *n*, *n* = **all** for all requests, (user's requests default)
- **kn** kill **uucp** job number *n*
- **msys** report accessibility status of **sys** (**sys** = **all** for status of all systems)
- **Msys** like -**m**, but include time last status obtained and time of last successful transfer to **sys**
- **on** status of all requests older than *n* hours
- **O** report status using octal codes
- **q** report job numbers, control files & time oldest & youngest files queued for each system
- **rn** set last modified time of job *n* to current time
- **ssys** status of requests logged with **sys**
- **uuser** status of requests from **user**
- **yn** status of all requests younger than *n* hours

UUTO - Public UNIX-to-UNIX File Copy

% **uuto** [*options*] *files destination*

Options:

- **m** mail when copy is completed
- **p** copy *files* to spool directory before transmit

UUX - Remote UNIX Command Execution

% **uux** [*option*] *cmd*

Option:

- **uux**'s **stdin** becomes *cmd*'s **stdin**
- **j** reverse action of **JOBNO** envir. variable
- **mfile** report transfer status to *file*, mail when complete if no *file* specified
- **n** don't notify user

VAL - Validate SCCS Files

% **val** [*options*] *files*

command lines read from **stdin** if *files* is -

Options:

- **mtext** *text* is compared with value of **%M%** keyword
- **rsid** specify SCCS ID of version
- **s** suppress **stdout** output
- **ytext** *text* is compared with value of **%Y%** keyword

VAX - Returns True Exit Status if **VAX** - 11/750 or 780

% **vax**

VC - Version Control

% **vc** [*options*] [*control_statements*]

Options:

- **a** replace keywords in all text lines
- **cc** change control char to *c* (: default)
- **s** don't print warning messages
- **t** ignore chars from beginning of line to first tab

Control Statements:

- :asg** *keyword* = *value*
assign *value* to *keyword*
- :ctl** *c* change control char to *c*
- :dcl** *keywords*
declare *keywords*; comma separated
- :err** *message*
print *message* on **stderr**;
halt execution; return exit code 1
- :if** *condition* <input lines>
:end
if *condition* true copy input lines to **stdout**

Operators:

- = equal
- != not equal
- & and
- pipe or
- > greater than; unsigned integers only
- < less than; unsigned integers only
- () logical groupings
- not invert value of condition

- :msg** *message*
print *message* on **stderr**
 - :off** turn keyword replacement off
 - :on** turn keyword replacement on
 - ::text** remove leading control chars; substitute values for keywords
- keywords* ≤ 9 alphanumeric, no blanks or spaces
values ASCII or numeric strings

VEDIT - Screen Editor for Beginners

% **vedit** [*options*] [*files*]

Options:

- + *pos* position file at *pos* (end of file default)
 - **l** set options appropriately for editing LISP
 - **r** retrieve last saved version of *file* after system or editor crash (list of all saved files default)
 - **R** read-only mode (same as **view**)
 - **t** *tag* edit file containing *tag* and position editor at its definition
 - **wn** set default window size to *n*
 - **x** create or edit encrypted file
- pos* any editor command not containing a space

VI - Screen Editor

% **vi** [*options*] [*files*]

Options:

- + *pos* position file at *pos* (end of file default)
 - **l** set options appropriately for editing LISP
 - **r** retrieve last saved version of *file* after system or editor crash (list of all saved files default)
 - **R** read-only mode (same as **view**)
 - **t** *tag* edit file containing *tag* and position editor at its definition
 - **wn** set default window size to *n*
 - **x** create or edit encrypted file
- pos* any editor command not containing a space

VIEW - Read-Only Screen Editor

% **view** [*options*] [*files*]

Options:

- + *pos* position file at *pos* (end of file default)
- **l** set options appropriately for editing LISP
- **r** retrieve last saved version of *file* after system or editor crash (list of all saved files default)
- **t tag** edit file containing *tag* and position editor at its definition
- **wn** set default window size to *n*
- **x** create or edit encrypted file
- pos* any editor command not containing a space

VPR - Spooler for Versatec Printer

% **vpr** [*options*] [*files*]

stdin read if no *files* specified

Options:

- **c** print from a copy of the *files*
- **f file** use *file* as dummy file name for reporting completion to user
- **m** send mail when printing complete
- **n** don't mail after printing done (default)
- **p** [- **e file**]
 - use **vplot** to output files by **graph**
 - **e** output scan converted raster *file*
- **r** remove *files* after spooling

WAIT - Wait for All Background Processes to Complete

% **wait**

WC - Count Lines, Words and Characters

% **wc** [*options*] [*files*]

stdin read if no *files* specified

Options:

- **c** output character counts
- **l** output line counts
- **w** output word counts

WHAT - Print SCCS Identifying Information (%Z% Value)

% **what** [*option*] *files*

Option:

- **s** stop after finding first occurrence of pattern

WHO - Who is on the System

% **who** [*options*] [*file*] [**am I**]

Options:

- **a** turn all options on
- **b** list time and date of last reboot
- **d** list expired processes not respawned by **init**
- **H** print column headings above output
- **l** list lines available for **login**
- **p** list active processes spawned by **init**
- **q** only list # of current users and their names
- **r** list info on run-level of **init** process
- **s** list current users' name, line and time logged in (default)
- **t** show last time **date** changed clock
- **T** list info on state of terminal
- **u** long list of info on logged in users
- am I** (or **am I**) outputs **who** you are logged in as
- file* read instead of **/etc/utmp** for login information

XARGS - Construct Argument List and Execute

% xargs [*options*] [*cmd* [*initial_args*]]

Options:

- **eof** set end of file string (underscore default)
- **ireplace** *cmd* executed with occurrences of *replace* in **stdin** replaced by *initial_args*
- **ln** *cmd* executed for each *n* lines of arguments
- **nn** *cmd* executed with up to *n* arguments
- **p** prompt user whether each *cmd* invocation is to be executed (**y** confirms execution)
- **sn** max size of any argument list is *n* characters
- **t** trace; executed *cmds* output to **stderr**
- **x** stop if any argument list greater than size

YACC - Yet Another Compiler Compiler

% yacc [*options*] *file*

Options:

- **d** #defines for token names and codes to **y.tab.h**
- **l** no #line constructs in **y.tab.c**
- **t** include debugging code in **y.tab.c**
- **v** parse tables and grammar reports to **y.output**

SHELL

Special Characters

	pipe
;	command separator
&	run process in background
&&	only run following command if previous command completed successfully
	only run following command if previous command failed
'	enclose string to be taken literally
"	enclose string to have parameter and command substitution only
\	inline command execution
\	ignore special meaning of following character
?	match single character in filename
*	match 0 or more characters in filename
[chars]	match any of <i>chars</i> (pair separated by a - matches a range)

Input/Output

All of these operators may be preceded by an optional file descriptor. Defaults are shown in parentheses.

<file	use <i>file</i> as stdin
>file	use <i>file</i> as stdout
>>file	like > but append to <i>file</i>
<&n	duplicate input file descriptor from <i>n</i> (stdin)
>&n	duplicate output file desc. from <i>n</i> (stdout)
<&-	close stdin
>&-	close stdout
<<arg	treat line with <i>arg</i> as EOF on input
<<-arg	as above with leading tabs ignored

Parameters

\$n	use positional parameter <i>n</i>
\$*	all parameters
\$@	all parameters
"\$*"	equivalent to "\$1 \$2 ..."
"\$@"	equivalent to "\$1" "\$2" ...
\$#	number of positional parameters
\$-	options to shell or by set
\$?	value returned by last command
\$\$	process number of current shell
\$_	process number of last background command
\$HOME	home directory for cd command
\$IFS	field separators (space , tab , newline)
\$MAIL	name of a mail file, if any
\$MAILCHECK	check mailfiles for mail every <i>n</i> seconds (600 default)
\$MAILPATH	colon separated list of file names
\$PATH	command search path
\$PS1	primary prompt string (\$)
\$PS2	secondary prompt string (>)
\$_SHACCT	accounting file for user shell procedures
\$_SHELL	if <i>r</i> anywhere, set up restricted shell
name = val	set <i>name</i> to specified value
\$_{var}	use braces to delimit shell variable name
\$_{var - str}	use parameter <i>var</i> if set, otherwise <i>str</i>
\$_{var = str}	as above but set <i>var</i> to <i>str</i> also
\$_{var ? str}	use <i>var</i> if set, otherwise print <i>str</i> and exit
\$_{var + str}	use <i>str</i> if <i>var</i> set, otherwise use nothing

Note: using *var*: instead of *var* checks only if *var* is set:
using *var* checks if *var* is set and non-NULL

SHELL, continued

Control Commands

: null command; returns 0 exit status
start of comment; terminated by a newline
break [*n*] exit from enclosing **for** or **while** loop
case *string* **in** [*pat1* [*pat2*]...] *list*;;...**esac**
execute *list* with *pat* that matches *string*
continue [*n*]
do next iteration of enclosing **for** or **while**
exit [*n*] exit with return value *n*
for *name* [**in** *strings*] **do** *list* **done**
do *list* setting *name* to each string
funct () {*list*;}
define *funct*, body of *funct* is *list*
if *list1* **then** *list2* [**elif** *list3* **then** *list4*]...[**else** *list5*] **fi**
if executing *list1* returns successful exit status,
execute *list2* else ...
(*list*) execute *list* in a subshell
{*list*;} *list* is just executed
while *list1* **do** *list2* **done**
execute *list1* if last command in *list1* had a
successful exit status, execute *list2*; repeat
until last command in *list2* returns
an unsuccessful exit status
until *list1* **do** *list2* **done**
like **while** but negate termination test

Special Commands:

. *file* read and execute commands from *file*
cd [*arg*] change current directory to *arg*
echo [*args*] echo arguments
eval [*args*] evaluate *args* and execute result
exec [*args*] execute *args*
export [*names*]
export *names* to environment of commands
hash [-*r*] [*names*]
if no *args*, show info on remembered cmds;
if *names*, remember location of each *name*;
-*r* forget all remembered locations
newgrp [*args*]
same as **exec newgrp** *args*
pwd print working directory name
read [*names*]
read **stdin** and assign to *names*
return [*n*] exit with return value *n*; if no *n*, return
status of last command
readonly [*names*]
mark *names* read only; print list if no *names*
set -*options* [*args*]
set flags (see **sh**); *args* set positional
parameters
set +*options* [*args*]
unset flags (see **sh**)
shift [*n*] rename positional parameters; $\$n + 1 = \1 ...
(*n* defaults to 1)
test evaluate conditional expressions (see **test**)
times print accumulated process times
trap [*arg*][*sigs*]
execute *arg* if signal in *sigs* received
type [*names*]
show how shell would interpret each *name*
ulimit [*options*] [*n*]
if no *n*, print current limits
-*f* limit file sizes to *n* blocks (default)
-*p* change pipe size to *n*
umask [*num*]
set file creation permissions mask
to complement octal *num*
unset [*names*]
unset specified variables or functions
wait [*n*]
wait for process *n*; if no *n*, wait for all children

Examples, continued

man spell
print Unix user's manual page for a command

mkdir /tmp/myjunk
make a new directory

mv file1 file2 /tmp
move files to specified directory

nroff file | hyphen
find hyphenated words ending lines in formatted file

pg +50 file
view file by screenful, starting at line 50

pr file | lp
paginate a file with default header, spool output

ps -l
print long listing of current processes, PID's and status

rm file
remove (delete) a file

rm -i junk[0-9]
*remove files **junk0** ... **junk9**, confirming first*

sort -o file file
sort a file in place

sort +3 -4 file
print file sorted only on fourth field

stty -a
print all terminal option settings

stty raw; prog; stty -raw
*set terminal to **raw** mode, run program, and restore mode*

tail -6 file
print the last six lines of a file

umask
print current file creation mask

umask 077
*set file creation mask, remove access by **group** and **other***

vi file
edit file using full screen editor

OTHER REFERENCES PUBLISHED BY:

Specialized Systems Consultants

P.O. Box 55549

Seattle, WA 98155

(206) FOR-UNIX

UNIX System Command Summaries

(Berkeley 4.2, XENIX, System III, System V)

C Library Reference for UNIX

C Reference Card

VI Reference

Fortran 77 Reference

MS-DOS Reference