



A/UX Reference Summary and Index

Release 3.0

LIMITED WARRANTY ON MEDIA AND REPLACEMENT

If you discover physical defects in the manuals distributed with an Apple product or in the media on which a software product is distributed, Apple will replace the media or manuals at no charge to you, provided you return the item to be replaced with proof of purchase to Apple or an authorized Apple dealer during the 90-day period after you purchased the software. In addition, Apple will replace damaged software media and manuals for as long as the software product is included in Apple's Media Exchange Program. While not an upgrade or update method, this program offers additional protection for up to two years or more from the date of your original purchase. See your authorized Apple dealer for program coverage and details. In some countries the replacement period may be different; check with your authorized Apple dealer.

ALL IMPLIED WARRANTIES ON THE MEDIA AND MANUALS, INCLUDING IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, ARE LIMITED IN DURATION TO NINETY (90) DAYS FROM THE DATE OF THE ORIGINAL RETAIL PURCHASE OF THIS PRODUCT.

Even though Apple has tested the software and reviewed the documentation, **APPLE MAKES NO WARRANTY OR REPRESENTATION, EITHER EXPRESS, OR IMPLIED, WITH RESPECT TO SOFTWARE, ITS QUALITY, PERFORMANCE, MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE. AS A RESULT, THIS SOFTWARE IS SOLD "AS IS," AND YOU, THE PURCHASER, ARE ASSUMING THE ENTIRE RISK AS TO ITS QUALITY AND PERFORMANCE.**

IN NO EVENT WILL APPLE BE LIABLE FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES RESULTING FROM ANY DEFECT IN THE SOFTWARE OR ITS DOCUMENTATION, even if advised of the possibility of such damages. In particular, Apple shall have no liability for any programs or data stored in or used with Apple products, including the costs of recovering such programs or data.

THE WARRANTY AND REMEDIES SET FORTH ABOVE ARE EXCLUSIVE AND IN LIEU OF ALL OTHERS, ORAL OR WRITTEN, EXPRESS, OR IMPLIED. No Apple dealer, agent, or employee is authorized to make any modification, extension, or addition to this warranty.

Some states do not allow the exclusion or limitation of implied warranties or liability for incidental or consequential damages, so the above limitation or exclusion may not apply to you. This warranty gives you specific legal rights, and you may also have other rights which vary from state to state.

 Apple Computer, Inc.

© 1992, Apple Computer, Inc., and UniSoft Corporation. All rights reserved.

Portions of this document have been previously copyrighted by AT&T Information Systems and the Regents of the University of California, and are reproduced with permission. Under the copyright laws, this manual may not be copied, in whole or part, without the written consent of Apple or UniSoft. The same proprietary and copyright notices must be affixed to any permitted copies as were affixed to the original. Under the law, copying includes translating into another language or format.

You may use the software on any computer owned by you, but extra copies cannot be made for this purpose.

The Apple logo is a registered trademark of Apple Computer, Inc. Use of the "keyboard" Apple logo (Option-Shift-k) for commercial purposes without the prior written consent of Apple may constitute trademark infringement and unfair competition in violation of federal and state laws.

Apple Computer, Inc.
20525 Mariani Avenue
Cupertino, CA 95014-6299
(408) 996-1010

Apple, the Apple logo, A/UX, ImageWriter, LaserWriter, and Macintosh are trademarks of Apple Computer Inc., registered in the United States and other countries.

B-NET is a registered trademark of UniSoft Corporation.

DEC and VT102 are trademarks of Digital Equipment Corporation.

Diablo and Ethernet are registered trademarks of Xerox Corporation.

Electrocomp 2000 is a trademark of Image Graphics Inc.

Hewlett-Packard 2631 is a trademark of Hewlett-Packard.

IBM is a registered trademark of International Business Machines Corporation.

NFS is a trademark of Sun Microsystems, Inc.

PostScript and TranScripts are trademarks of Adobe Systems Incorporated, registered in the United States.

UNIX is a registered trademark of UNIX Systems Laboratories, Inc.

Simultaneously published in the United States and Canada.

Mention of third-party products is for informational purposes only and constitutes neither an endorsement nor a recommendation. Apple assumes no responsibility with regard to the performance or use of these products.

A/UX Reference Summary and Index

Contents

About This Manual

Commands by Function

Command Synopses

Index

About This Manual

This manual is one of the secondary A/UX reference manuals. It supplements each of the three primary manuals: *A/UX Command Reference*, *A/UX Programmer's Reference*, and *A/UX System Administrator's Reference*. The *A/UX Reference Summary and Index* is designed to help you find information in the other reference volumes.

The reference books cited above, from which *A/UX Reference Summary and Index* is derived, are encyclopedic collections of manual pages, not narrative or tutorial works. They provide complete technical information about all the programs, utilities, and standard file formats included with your A/UX system.

Because all of these reference manuals are not intended to be tutorials or learning guides, they should not be the first A/UX books you read. If you are new to A/UX or are unfamiliar with a specific functional area (such as the Finder), you should first read *A/UX Essentials* and the other A/UX user guides. After you have worked with A/UX, the reference manuals can help you understand new features or refresh your memory about features you already know. This manual, *A/UX Reference Summary and Index*, further assists you by providing several ways to find exactly the information you want.

Locating information in the reference manuals

You can locate information in the reference manuals with the help of the following sections in each of the primary reference manuals:

- **Table of contents.** Each reference manual contains one general table of contents for the entire manual. Located at the beginning of each new section of manual pages is a detailed table of contents. (If a section must span from one binder to another, a tailored table of contents is provided for each of the subdivisions.) The general table of contents lists the sections covered in the complete manual. The detailed table of contents lists the manual pages contained within one section (or section subdivision) along with a brief description of the A/UX provision that is covered in each manual page.
- **Query commands.** The `man`, `what is`, and `apropos` commands display on-screen all the information contained in a manual page or just the information in the "Name" section of one or more manual pages that

satisfy a search criterion. A section that appears later in this preface, “Using the Online Documentation,” tells you how to use the online versions of the manual pages.

- This book, *A/UX Reference Summary and Index*. This separate manual is considered part of the A/UX set of reference manuals, but it is not a “standard” resource like the other reference materials. Its primary purpose is to help you locate the manual pages you need. From its summaries, you might also occasionally find all the information you required. It contains the following subsections:
 - “Commands by Function.” This subsection classifies the A/UX user and system administrator commands by the general or most important function each performs. The summary gives you a broader view of the commands that are available and the context in which each is often used. Each command is mentioned just once in this listing.
 - “Command Synopses.” This subsection is a compact collection of syntax descriptions for all of the commands in *A/UX Command Reference* and in *A/UX System Administrator’s Reference*. It may help you find the syntax of commands more quickly when the syntax is all you need.
 - “Index.” The index lists key terms associated with A/UX subroutines and commands. These key terms can help you locate the manual page you need as you browse through keyword-related commands and subroutines.

Using this manual

This manual contains three parts, separated by tab dividers. Each part is to be used differently.

Commands by function

With A/UX you are confronted with a multitude of commands. To help you sort them out, the first section of this book is a command summary. It groups commands together according to the functions they perform. Each command is mentioned just once in the summary, in accordance with its general, or most important, function. This way you get a bird’s-eye view of the overall command capabilities of A/UX.

This command summary mentions all the user commands in *A/UX Command Reference* and *A/UX System Administrator’s Reference*. The commands are categorized under headings such as “Logging In and Logging Out” and

“Formatting Text Into Pages for Printing.”

To locate the commands for a function or task that interests you, first consult the list of major categories given at the start of the summary section. It lists the principal heads under which commands are grouped. When you find the appropriate major category, turn to the starting page indicated. There you will find lists of A/UX commands for functions and tasks within the major category.

A mention of a command in this summary typically looks like:

change login password.....passwd

To change a password, you are directed to the `passwd` command. The brief function description (“change login password” in this example) applies to the command (`passwd`).

Command synopses

Most tasks require that you enter information on the command line after the name of the command, such as flag options that modify the behavior of the command. Often you must supply other arguments as well, such as the names of files. Each man page includes a syntax synopsis that helps you construct command lines.

This section gathers into one place all the synopsis sections from sections 1, 1M, 5, 6 and 8 of *A/UX Command Reference*, *A/UX Programmer’s Reference*, and *A/UX System Administrator’s Reference*. It helps you find the syntax of commands quickly and is helpful when the syntax is all you need to see. The synopses are presented in alphabetical sequence by command name. Left and right guide words at the top of each page indicate the first command and the last command covered on that page.

Index

The A/UX references contain a large amount of information, so finding a specific fact in them can be a daunting task. The Index section is designed to help you locate specific man pages by providing cross-references to them from a variety of topic headings.

Most manual pages are indexed under more than one topic heading; for example, `lorder(1)` is included under “archive files,” “sorting,” and “cross-references.” This way you are more likely to find the reference you are looking for on the first try.

The Index section works like an ordinary index, except that a short description is included along with each reference. This description, plus the index topic help you quickly determine whether a reference contains the information you want. Once you determine that you want to view a manual page, the parenthetical section number helps you find the correct book in which to look. Suppose you have located a reference to `adduser(1M)`. The sections that appear in each of the reference manuals are listed on the book front cover and spine. Section 1M appears on the spine of *A/UX System Administrator's Reference*. Accordingly, that is where you can find the full text of `adduser(1M)`. Also, see “Using the Online Documentation,” later in the preface.

The key terms in this index were constructed by examining the meaning and usage of the A/UX manual pages. They are designed to be more discriminating and easier to use than the traditional permuted index, which mechanically lists keywords found in the manual page NAME sections.

Visual conventions for the A/UX reference manuals

A/UX books follow specific styling conventions. For example, words that require special emphasis appear in specific fonts or styles. This section describes the conventions used in all the A/UX reference books.

The Courier font

Throughout the A/UX reference manuals, words that appear on the screen or that you must type exactly as shown are in the Courier font.

Here's an example:

Type `date` on the command line and press RETURN.

This instruction means that you should type the word “date” exactly as shown, then press the RETURN key.

After you press RETURN, text such as this will appear on the screen:

```
Fri Nov  1 11:15:43 PST 1991
```

In this case, the Courier font is used to represent exactly what appears on the screen.

All A/UX manual page names are shown in the Courier font. For example, `ls(1)` indicates that `ls` is the name of a manual page that occurs in Section 1. More information about the use of the Courier font in manual pages is given in “Styling of A/UX Command Elements” later in this preface.

Font styles

Italics are used to indicate that a word or set of words is a placeholder for part of a command line. Here is a sample command syntax illustration:

```
cat file
```

The italicized term *file* is a placeholder for the name of a file. If you wanted to display the contents of a file named `Elvis`, you would type the filename `Elvis` in place of *file*. In other words, you would enter

```
cat Elvis
```

Styling of A/UX command elements

A/UX commands are entered in accordance with their command syntax. A typical A/UX command line includes the command name first, followed by options and arguments. For example, here is an illustration of the syntax for the `wc` command:

```
wc [-l] [-w] file...
```

In this syntax illustration, `wc` is the command, `-l` and `-w` are options, and *file* is an argument.

A “command option” modifies the action of a command, often by changing its mode of operation (such as read mode or write mode).

An “argument” is any element that follows the command name. Command arguments other than command options typically specify the objects upon which the command should act. You often supply the names of files that you want a command to process, so *file* is frequently the last element in syntax illustrations.

Brackets and ellipsis characters in a syntax illustration should be considered part of a syntax notation. This is represented by the use of body font instead of Courier for these characters. Their font treatment tells you that you are not supposed to type these characters as part of the command line. Their meaning as a syntax notation is described next.

The brackets enclose an optional item or a group of optional items. If an optional item has constituent parts that are also optional, these parts are themselves enclosed in brackets, as in this syntax illustration:

```
lpr [-i [numcols]]
```

This syntax illustration shows that the indent (`-i`) command option can be followed by the number of columns to indent the printed page. It also shows that

you can omit the number of columns; if you do, the `lpr` command uses the default indent value.

An ellipsis (...) follows an argument that can be repeated any number of times on a command line. If the ellipsis follows a bracketed group of items, the group of items can be repeated any number of times on the command line.

Mutually-exclusive command options cannot be specified within the same command-line request. For commands that have mutually-exclusive options, two or more command-line syntaxes are offered:

```
pax -r[other-option-for-archive-reading]...  
pax -w[other-option-for-archive-writing]...
```

Outside of syntax illustrations, command options are shown with a leading hyphen also in the Courier font. When you supply multiple command options in an actual command line, only one leading hyphen is normally required. For example the following command line contains two options, `-r` and `-f`:

```
pax -rf /dev/rfloppy0
```

In the example, the `-f` option (pronounced “minus f”) is entered without its own hyphen, even though when mentioned in running text it appears with a leading hyphen.

Using the online documentation

In addition to the paper documentation in the reference manuals, A/UX provides several ways to search and read the contents of each manual page from your A/UX system. An advantage to the online version of the documentation is that the computer performs the work of filtering out (or skipping) all the manual pages other than the one you specifically queried. The only prerequisite is that you already know its name (or a proper search string). However, you don't have to know how manual pages are organized by section numbers and by book titles.

To display a manual page on your screen, enter the `man` command followed by the name of the manual page you want to see. For example, to display the manual page for the `cat` command, including its description, syntax, options, and other pertinent information, you would enter

```
man cat
```

After the first screen of the text of a manual page appears, you can display subsequent screens of the text with each press of the SPACE BAR, until you reach

the end of the man page. To display subsequent text one line at a time, press RETURN instead of the SPACE BAR. By pressing Q, you can quit the man command before viewing all of the manual page.

To display the descriptive information in the “Name” section of any manual page, enter the `what is` command followed by the name of the provision you want described. In the following example, the command prompt is the percent sign, and the provision that is being queried is the `ls` command:

```
% whatis ls
ls(1)          - lists the contents of a directory
% █
```

To display a list of all manual pages whose “Name” sections contain a given keyword or string, enter the `apropos` command followed by a search word or search string enclosed in double quote characters. The names of A/UX provisions are listed on separate lines along with the descriptive information in the “Name” section of the manual page that describes those provisions. Sometimes several A/UX provisions are listed on the same line. In those cases, several A/UX provisions are described on a single manual page. You can tell which of these names is the formal name for the manual page because it will be followed by parentheses and an enclosed section number. In the following example, the command prompt is the percent sign, and the A/UX provisions that are queried are those which are described in manual pages whose “Name” section contains the word “tape”:

```
% apropos tape
mt(1)          - magnetic tape manipulating program
frec(1M)       - recover files from a backup tape
mtio(7)        - interface conventions for magnetic tape devices
tc(7)          - Apple Tape Backup 40SC device driver
% █
```

These documentation query commands are described more fully in the manual pages `man(1)`, `what is(1)`, and `apropos(1)` in *A/UX Command Reference*.

For more information

To find out where you need to go for more information about how to use A/UX, see *Road Map to A/UX*. This guide contains descriptions of each A/UX guide and ordering information for all the guides in the A/UX documentation suite.

Commands by Function

This section lists all A/UX commands categorized by the functions they perform. The major functional categories appear in bold type. These major categories begin on the pages shown following:

Accessing the System and Its Help Resources	4
Managing Files and Directories	6
Controlling the User Interface	10
Controlling How Commands Are Run	11
Managing Processes as They Run	13
Generating Command Lines	13
Communicating	14
Playing Games	16
Processing Text as Records Within a Database	17
Processing Structured or Unstructured Text	18
Processing Text to Produce Printed Documents	20
Processing Plotter Drawings	23
Writing Shell Programs	24
Programming	25
Administering Your System	28

Each category includes one or more subcategories. While the category appears in bold text, the subcategory appears in plain text. Under each subcategory are the related functions.

To find a command that can count words in a file, you might follow this sequence of actions:

1. Locate “Processing Structured or Unstructured Text” (page 18) as the most appropriate main category.
2. Turn to page 18 and browse through the pages following until you locate “Report Occurrences of Words and Letters” as the next most appropriate subcategory.
3. Locate the phrase *word count* as the function desired.
4. Locate the command `wc` across from the phrase *word count*.

Once you have found a command likely to perform a desired function, you can get further information about that command by referring to the *A/UX Command Reference* and *A/UX System Administrator’s Reference*. An even faster way to

locate information is to use the online help provisions of A/UX. (See “Using the Online Documentation” in the preface.)

Normally, the names of manual page subdocuments are the same as the names of the commands they describe. This is not true when a manual page subdocument describes more than one command. An example is `rmdir`, which is described on the `rm(1)` manual page. You can use the `what is` command to help you locate the actual manual page for more information about a given command. Another way to locate the `rm(1)` manual page for commands like `rmdir` is to enter `rmdir` as the argument to the `man` command. The `man` command automatically locates the `rm(1)` manual page and displays it.

The categories are listed in no special order. Generally, the order of subcategories is alphabetical. The order of command names and descriptions is generally alphabetical as well, based on the command name.

The uses of some commands fit several categories. For example, the command fits equally well within two categories, “Performing Arithmetic Calculations” and “Interpreting Command Lines.” To make the summary brief, however, each command is listed under only one classification. Note also that rarely-used commands are interspersed among frequently-used commands, falsely suggesting that each command is equally useful.

Also, the manual pages cited for a specific category may not provide an adequate overview of a given topic. For example, the cited commands for “Directing Data To and From Files,” `tee` and `cat`, do not provide adequate guidance about input and output redirection. Redirection is best described in other A/UX books, such as *A/UX Shells and Shell Programming*. (The following manual pages from the *A/UX Command Reference* also describe input and output redirection: `sh(1)`, `csh(1)`, and `ksh(1)`.)

Finally, certain categories are necessarily nondescript, such as “Using Devices.” Since you are using devices whenever you use A/UX, all commands could have been placed in this category. However, only those commands more concerned with manipulating devices than manipulating files or data were placed in this category. The chief concern for the choice of category titles was finding titles that are clear when considered with the other categories. Accordingly, the category titles taken by themselves often fail to describe precise sets of commands.

If you are confident using Macintosh applications but uncertain about the added value that A/UX can provide, the following categories are likely to interest you.

Accessing the System and Its Help Resources. This topic includes many subcategories of general interest and commands that are likely to be used with medium frequency. Of these, the most frequently used commands are the commands used to obtain online help.

Managing Files and Directories. This topic includes the most frequently used commands in the system. When managing large numbers of folders and files,

A/UX command lines may be preferable to Finder operations. For example, command lines can be used to manipulate files in a nested folder without having to use a prior operation to “open” the nested folder.

Controlling How Commands Are Run. Among the commands listed are those that allow you to schedule commands to run in a recurring fashion, or in a time-delayed fashion.

Communicating. This topic includes commands that support the popular UNIX utility for electronic mail. To use the mail facilities of A/UX optimally, you could even create customized scripts that automatically start up according to specific dates and times. (See “Writing Shell Programs” and “Controlling how Commands are Run.”)

Processing Structured or Unstructured Text. The editors are frequently used to edit database style tables, such as `/etc/passwd`, as well as to edit document text. `TextEditor` is the editor of choice if you wish to take advantage of your Macintosh skills. The `grep` command is a frequently used A/UX utility that displays lines in any text file containing a string or substring you specify.

Processing Text Records. Within certain limits, the commands listed here can process information from files generated with Macintosh spreadsheet and database applications once they are saved as text. Another powerful provision, but one that is categorized differently from these, is `awk`. It is a high-level programming language used to write programs that process text or compile custom reports from field-structured text files.

Writing Shell Programs. The A/UX shell programming languages are frequently used to automate recurring tasks or to bind several related actions into an easily-invoked command script. The shells allow users to easily create new A/UX functions, extending the repertoire of existing programs in ways that fit the needs of a particular site. A number of the supplied A/UX programs are actually shell scripts, so they can be readily copied and customized.

Accessing the system and its help resources

Finding out about your network

displays a list of the active users from all of the systems on the
local network rwho
displays the host status of local machines ruptime
produces a login list for local machines (RPC version) rusers

Finding out about your system

displays a summary of the current system activity w
displays group memberships groups
displays identification information about the current system uname
displays information about the users on a system finger
displays login and logout times for each user of the system last
displays the system page size pagesize
displays user and group IDs and names id
provide truth values about processor type machid
reports a list of the users who are logged on to the system users
reports how long system has been up uptime

Finding out about your session

displays the value of variables set in the current
environment printenv
gets the login name logname
obtains the device filename for the terminal or CommandShell
window where it is invoked tty
prints the name of the working directory pwd
reports process status ps

Getting online help

displays the named manual page entries man
informs you of the current system activity who do
locates commands by keyword apropos
prints effective current user ID whoami
reports a brief description for the manual page entry specified what is
reports the directory path to a file by interpreting PATH and
alias settings which
reports the locations of the source, binary, and online help
files for a specified command where is
reports users who are currently logged in to the system who

Logging in and logging out

changes the login password passwd
logs in to a remote system rlogin
logs you into a new group newgrp
signs you on a terminal session login

Performing arithmetic calculations

desk calculator dc
prints the prime factor of a given number factor
processes an arbitrary-precision arithmetic language bc
rescales quantities according to a the unit of measure specified units

Using devices

blocks data to 8K for direct input to /dev/rmt/tcx tcb
clears the terminal screen clear
ejects a diskette from the drive eject
manipulates magnetic tape media mt
prepares data to be printed on the Apple ImageWriter II printer iw2
sets the modes for a terminal stty
sets the tab stops on a terminal tabs

Using time and date utilities

displays a calendar cal
displays and sets the date date
provides a reminder service calendar
reminds you when you have to leave leave

Managing files and directories

Changing file attributes

change the owner or group of a file chgrp
change the owner or group of a file chown
changes the permissions of a file chmod
sets attributes for Macintosh files, such as file type and
creator setfile
updates access and modification times of a file touch

Comparing files and directories

compares the contents of two directories dircmp
compares the difference between two large files that are too big
for diff to handle bdiff
compares three versions of a file diff3
compares two files or directories for any differences diff
compares two files cmp
merges three files into one merge
reports differences between two files or directories ucbsdifff
reports side-by-side differences between two files in a side-by-
side format sdiff
reports the differences between three files ucbsdifff3
sums and counts the characters within the files of the given
directories sumdir

Compressing and encrypting files

compress and expand files	pack
compress and expand files	pcat
compress and expand files	unpack
compress and uncompress files	ccat
compress and uncompress files	compact
compress and uncompress files	uncompact
compress files and directories as well as expand them; support concatenation, browsing, and file- comparing operations upon compressed files	compress
compress files and directories as well as expand them; support concatenation, browsing, and file- comparing operations upon compressed files	compressdir
compress files and directories as well as expand them; support concatenation, browsing, and file- comparing operations upon compressed files	uncompress
compress files and directories as well as expand them; support concatenation, browsing, and file-comparing operations upon compressed files	uncompressdir
compress files and directories as well as expand them; support concatenation, browsing, and file-comparing operations upon compressed files	zcat
compress files and directories as well as expand them; support concatenation, browsing, and file-comparing operations upon compressed files	zcmp
compress files and directories as well as expand them; support concatenation, browsing, and file-comparing operations upon compressed files	zdiff
compress files and directories as well as expand them; support concatenation, browsing, and file-comparing operations upon compressed files	zmore
encodes and decodes passwords	crypt

Copying files and directories

converts a file in one storage format to a different storage format ... `fcvt`
converts and copies a file `dd`
copies files between two systems `rcp`
copies files to or from a `cpio` archive `cpio`
copies files to or from a `tar` archive `tar`
copies files to or from a `tp` archive `tp`
copies files to or from an archive in an IEEE format `pax`
copies files `cp`
makes links `ln`
splits a file into a specified number of pieces `split`
splits files into sections `csplit`

Creating, renaming, and removing files and directories

creates a directory `mkdir`
moves or renames files `mv`
remove files or directories `rm`
remove files or directories `rmdir`

Directing data to and from files

catenates and displays the contents of files `cat`
transcribes data `tee`

Displaying filenames and file status

calculates a checksum `sum`
determines the type of a file `file`
lists the contents of a directory `ls`
reports version number of files `version`

Finding files

finds files find

Finding out about your file system

reports the used and unused storage capacity for a file system df

summarizes disk usage du

Looking at files

displays the first few lines of a file head

displays the last part of a file tail

show the contents of a file in display-size chunks more

show the contents of a file in display-size chunks page

shows the contents of a file in display-size chunks pg

Printing files

cancel print requests spooled through the lp command cancel

queries the print spooler for progress information lpq

removes jobs from the line printer spooling queue for a Berkeley
file system (4.2) lprm

spools print requests to printers lp

spools print requests to printers lpr

Controlling the user interface

Choosing session preferences

changes the default login shell chsh
logs you in to A/UX by using a graphical user interface Login
manages command-interpretation windows and
moderates access to the A/UX console
window CommandShell

Customizing the Macintosh system for one user account

create a personal System Folder systemfolder

Interpreting command lines

discontinues a csh login session (a function built into ksh) logout
discontinues command interpretation in the current shell (as a
function built into the shell) exit
evaluates its arguments as a command line a specified number
of times (as a function built into the shell) repeat
evaluates its arguments as a command line and then exits shell (a
function built-into the shell) exec
evaluates its arguments as a command line as a function built into
the shell eval
manages the layering of multiple shells sh1
runs the Bourne shell sh
runs the C shell, a command interpreter with C-like syntax csh
runs the Korn shell, an enhanced command interpreter that is
backward-compatible with the Bourne shell (sh)
..... ksh

Launching Macintosh applications

runs a Macintosh binary application in A/UX launch

Controlling how commands are run

Delaying a command or part of a shell script

suspends the system for a specified interval of time sleep

Establishing the environment for a Macintosh application

changes or displays the fields of the 'SIZE' resource of
a file changesize

convert between Macintosh encoding and International
Standards Organization (ISO) encoding isotomac

convert between Macintosh encoding and International
Standards Organization (ISO) encoding mactois

Establishing the execution environment for a command

..... ulimit

changes the current working directory as a function built into the
shell cd

changes the root directory for a command chroot

displays or resets default file permissions as a function built into
the shell umask

executes a command at low priority nice

generates y entries in response to requests for input yes

invokes to a shell on a remote system remsh

runs a command so that it can continue to run even after your
session has ended nohup

sets the environment for command execution env

Interpreting command lines while maintaining an audit trail

starts a shell that records terminal input and output script

Setting a time at which to run a command

aids in the use of the `cron` process scheduling program..... `crontab`
run commands at a later time `at`
run commands at a later time `batch`
runs the clock daemon `cron`

Managing processes as they run

Signaling and terminating processes

removes interprocess communications facilities `ipcrm`
terminates a process `kill`

Timing the duration of a process

prints the elapsed time during the execution of a command `time`
reports the elapsed, user, and system time during the execution
of a command `timex`

Generating command lines

Constructing and executing command lines

builds arguments based on the standard input, passing them in
batches to the specified command which is executed
enough times to deplete all the arguments `xargs`
passes its arguments in batches to a command that is run once
per every batch `apply`

Constructing command lines using Macintosh dialog boxes

builds command lines interactively `cmdo`

Communicating

Communicating with other users

displays local news items news
displays the mail header lines in your mailbox from
enables and disables notification of mail by comsat biff
enables you to send and receive messages electronically mailx
permits or denies the receipt of messages mesg
send mail to users or read mail mail
talks to another user via the terminal talk
writes to all users wall
writes to another user write

Using AppleTalk

allows you to choose a default printer on the AppleTalk
internet at_cho_prn
displays status information from an AppleTalk device atstatus
looks up network-visible entities (NVEs) registered on the
AppleTalk network system atlookup
transfers data to a printer by using AppleTalk protocols atprint

Using TCP/IP

assigns a serial line to a network interface slip
communicates with another host via the TELNET protocol telnet
displays the status of machines on the local network (RPC version).... rup
distributes remote files rdist
responds to requests to use the DARPA Trivial File Transfer
Protocol tftpd
transfers files by using the DARPA Internet File Transfer Protocol
(FTP) ftp
transfers files via the Trivial File Transfer Protocol (TFTP) tftp
writes to all users over a network rwall

Using UUCP

controls uucp jobs and provides status information	uustat
copies files from one system to another system	uucp
displays information about uucp file transfers	uulog
displays the names of systems to which uucp and cu can connect	uuname
displays the service grades that are available on your system	uuglist
provide an easy interface to the uucp command, using the public directories	uupick
provide an easy interface to the uucp command, using the public directories	uuto
runs a command on a remote system	uux
sends a file to a remote host	uuse

Using other communications tools

encodes and decodes a binary file	uudecode
encodes and decodes a binary file	uuencode
establishes a connection to a remote system	tip
establishes an interactive connection with another system	cu
invokes the Kermit file-transfer program	kermit
runs login on a dial-up line	ct
updates files between two machines	updater

Playing games

animates raindrops	rain
converts Arabic numerals to English	number
generates a maze	maze
gives associative knowledge tests on various subjects	quiz
play the game of tic-tac-toe	cubic
play the game of tic-tac-toe	ttt
plays the game of autorobots	autorobots
plays the game of backgammon	back
plays the game of black jack	bj
plays the game of chase	chase
plays the game of craps	craps
plays the game of cribbage	cribbage
plays the game of fortune telling	fortune
plays the game of Go Fish''	fish
plays the game of growing worm	worm
plays the game of hangman	hangman
plays the game of hunt-the-wumpus	wump
plays the game of life	life
plays the game of Mastermind	mastermind
plays the game of moo	moo
plays the game of robots	robots
plays the game of Space Invaders (A/UX version)	aliens
plays the game of trek	trek
plays the game of twinkle, twinkle little stars	twinkle
plays the game of worms	worms
provides arithmetic problems	arithmetic
simulates a punched card corresponding to a text argument	bcd

Processing text as records within a database

Processing sorted text records

combines (joins) two relational files join
reports repeated lines in a file uniq
selects or rejects lines common to two sorted files comm

Processing text records and fields

cuts out selected fields of each line of a file cut
merges lines of several files or subsequent lines of one file paste
removes columns from a file colrm
sorts or merges files sort

Processing structured or unstructured text

Editing text

edit text e
edit text ed
edit text ex
edit text red
edits big files bfs
invokes the screen-oriented (visual) display editor vedit
invokes the screen-oriented (visual) display editor vi
invokes the screen-oriented (visual) display editor view
lets you edit files interactively through mouse and menu
operations TextEditor

Generating custom text transformations

edits a stream of data sed
generates an encryption key makekey
scans a file for lines that match a specific pattern awk
translates characters tr

Printing poster-size text

generates a large banner banner7
generates a poster banner

Processing tabbed text

changes the format of a text file newform
expand tabs to spaces, and vice versa expand
expand tabs to spaces, and vice versa unexpand

Reporting occurrences of words or letters

counts characters, words, and lines in a file	wc
finds references in a bibliography	lookbib
reports character frequencies in a file	freq
search a file for a specific pattern	egrep
search a file for a specific pattern	fgrep
search a file for a specific pattern	grep

Processing text to produce printed documents

Filtering printer motions from text for display purposes

filter text containing printer control sequences for a DASI terminal	300
filter text containing printer control sequences for a DASI terminal	300s
filters special underlining sequences imbedded in text for use at a display device	ul
filters text containing printer control sequences a page at a time	4014
filters text containing printer control sequences for the DASI terminal	450
filters text containing printer control sequences for use at a display device	col
filters text for vintage display devices	greek
filters <code>nroff</code> output for terminal previewing	colcrt
interprets <code>troff</code> output for use at a vintage display device	tc

Formatting text into pages for printing

converts text files to format for printing	enscript
converts <code>troff</code> intermediate format to POSTSCRIPT format	psdit
formats a file through <code>troff</code> so it can be printed on a POSTSCRIPT printer	psroff
formats and typesets files	troff
formats documents that contain <code>nroff</code> and <code>mm</code> macro formatting requests	mm
formats text for a print device	pr
formats text for a specific phototypesetter	otroff
invokes the Autologic APS-5 phototypesetter <code>troff</code> post-processor	daps
prints out all records in a bibliographic database	roffbib
text formatter	nroff
typeset documents that contain <code>troff</code> and <code>mm</code> or <code>mv</code> macro-formatting requests	mmt
typeset documents that contain <code>troff</code> and <code>mm</code> or <code>mv</code> macro-formatting requests	mvt

Preparing text with troff markup

analyzes the surface characteristics of documents	style
builds an inverted index for a bibliography	indxbib
creates a subject-page index for a document	ndx
creates or extends a bibliographic database	addbib
find spelling errors	hashcheck
find spelling errors	hashmake
find spelling errors	spell
find spelling errors	spellin
finds and inserts literature references in documents	refer
generates a list of subjects from documents	subj
generates a permuted index	ptx
locate wordy sentences in a document	diction
locate wordy sentences in a document	explain
sorts bibliographic database	sortbib

Preprocessing subsidiary markup within troff markup

eliminates the source commands from nroff input	soelim
folds long lines for finite-width output device	fold
format mathematical text for troff	eqn
formats mathematical text for nroff	neqn
invokes a simple text formatter	fmt
invokes a pic preprocessor for drawing graphs	grap
prepare constant-width text for oTROFF	cw
preprocesses troff files that contain drawings	pic
processes a file through a line numbering filter	nl
produces single spaced output	ssp
table formatter for nroff or troff	tbl

Processing troff-related markups for special purposes

check documents formatted with the mm macros checkmm
check documents formatted with the mm macros checkmm1
checks nroff/troff files checknr
finds hyphenated words hyphen
format mathematical text for troff checkeq
marks the differences between two files diffmk
prepare constant-width text for otroff checkcw
produces a cross-reference listing of macro files macref
removes nroff/troff, tbl, and eqn constructs deroff

Setting up device-specific fonts for use with troff

prepares troff description files makedev

Processing plotter drawings

Filtering plotter input for display purposes

interprets plotter instructions for use at a vintage display device `tplot`

Processing graphics

draws a graph `graph`

interpolates a smooth curve `spline`

Writing shell programs

Evaluating expressions to provide true or false results

evaluates conditions test
provides truth values false
provides truth values true

Evaluating math or string expressions

echoes its arguments echo
evaluates arguments as an expression expr
get part of a pathname basename
get part of a pathname dirname
parses command options getopt
reverses characters within each line of text rev

Performing input or output operations

posts a Macintosh alert box to query the user macquery
prints its arguments as a function built into the Korn (ksh)
shell print
queries the user for input query
queries terminfo database tput
reads one line from the standard input line

Programming

Using Macintosh development tools

compiles Macintosh resource files from source code rez
decompiles a resource file derez

Using other programming tools

assembles files by translating assembler mnemonics to object code as
compiles and interprets `bs` programs bs
compiles compilers (yet another compiler-compiler) yacc
compiles regular expressions with a file regcmp
converts an object file to Motorola S-record format hex
converts binary data to a displayable form in octal, decimal,
hexadecimal, or ASCII od
creates a shared library mkshlib
debugs and executes programs dbx
debugs executable programs adb
displays profile data prof
displays section sizes of common object files size
displays the symbol table of a common object file nm
finds the ordering relation for an object library lorder
finds the printable strings in an object or other binary file strings
generates C source code from a remote procedure call (RPC)
source file rpcgen
generates programs for simple lexical tasks lex
invokes the link editor for common object files ld
maintains a library of files in an archive ar
maintains, updates, and regenerates groups of files make
produces an assembly language listing for a specified file dis
receives and converts Motorola S-records from a port to a file rcvhex
sorts lines in a file topologically tsort
stores (saves) selected parts of an object file dump
strips symbol and line number information from an object file strip
swaps bytes in COFF files conv
symbolic debugger sdb

Using the C language

creates an error message file by massaging C source programs mkstr
debugs a C program ctrace
generates a C flowgraph cflow
generates a C program cross-reference cxref
improves spacing and indentation of C source files cb
indents and formats C program source indent
invokes a C program checker lint
invokes the C compiler cc
invokes the C language preprocessor cpp
maintains a tags file for a C program ctags
reports strings from C programs to implement shared strings xstr

Using the Fortran language

filters the output of Fortran programs for line printing fpr
interprets ASA carriage control characters asa
invokes the Extended Fortran Language efl
invokes the Fortran 77 compiler f77
splits f77 or efl files fsplit

Using unusual programming languages

processes macros for C and other languages m4
runs the SNOBOL interpreter sno

Using version management (RCS)

checks in RCS revisions ci
checks out RCS revisions co
compares RCS revisions rcsdiff
creates new RCS files or changes attributes of existing RCS files rcs
displays log messages and other information about RCS files rlog
displays RCS keywords and their values ident
merges two versions of an RCS file rcsmerge

Using version management (SCCS)

builds an RCS file from an SCCS file sccstorcs
changes the delta commentary of an SCCS delta cdc
combines SCCS deltas comb
compares two versions of an SCCS file sccsdiff
creates and administers SCCS files admin
displays information about an SCCS file prs
displays who has checked a Source Code Control System (SCCS)
file out for editing sact
gets a version of an SCCS file get
makes a delta (change) to an SCCS file delta
manipulates version control information inside a data stream vc
performs SCCS subsystem commands sccs
provides help information about SCCS commands and messages help
removes a delta from an SCCS file rmdel
reports identification information for a file what
undoes a previous get of an SCCS file unget
validate SCCS file val

Administering your system

AppleTalk network maintenance

- enables you to configure and display AppleTalk network interfaces `appletalk`
- exercises the AppleTalk network by sending packets to a named host `appleping`

Backing up your system

- copies blocks interactively `bcopy`
- copies System V File System-style file systems for optimal access time `dcopy`
- copy file systems with label checking `labelit`
- copy file systems with label checking `volcopy`
- create a `dump.bsd` archive by making copies of files from a given file system `dump.bsd`
- create a `dump.bsd` archive by making copies of files from a given file system `rdump`
- generates a fast incremental backup for System V file systems `finc`
- helps you with autorecovery administration `escher`
- recovers files from a backup tape `frec`
- retrieve files from within a `dump.bsd` archive into an existing file system `restore`
- updates autorecovery files `eu`
- updates important files for autorecovery purposes `eupdate`

Examining system status

- calls the error-logging daemon `errdemon`
- displays kernel name cache statistics `ncstats`
- extracts error records from a crash dump `errdead`
- prints system facts `pstat`
- processes a report of logged errors `errpt`
- terminates the error-logging daemon `errstop`
- turns on/off the reporting of extended errors `exterr`

File system maintenance

checks file-system consistency and interactively repairs the file system `fsck`
clears inodes `clri`
constructs a file system with 512-byte blocks `mkfs1b`
constructs a System V file system `mkfs`
creates an entry in the file-system table `fsentry`
debugs the file system `fsdb`
displays the current device name `devnm`
identifies processes using a file or file structure `fuser`
installs random inode generation numbers `fsirand`
lists file names and statistics for a System V file system `ff`
locates the filename associated with an i-number `ncheck`
makes a Berkeley 4.2 (UFS) file system `newfs`
makes a directory named `lost+found` to be used
by `fsck` `mklost+found`
mount and unmount file systems `mount`
mount and unmount file systems `umount`
reports the file-system type `fstyp`
reports the state of a file system `fsstat`
tunes a Berkeley 4.2 (UFS) file system `tunefs`
updates the superblock `sync`

Installing new software

installs files in specified directories `cpset`
installs A/UX software from specially prepared floppy disks ... `finstall`
places files in specified directories `install`

Kernel generation

creates an up-to-date kernel `autoconfig`
generates an up-to-date kernel `newconfig`
manipulates the files that determine the configuration of a
new kernel `newunix`
queries kernel files for configuration information `module_dump`
tunes kernel parameters for work-load optimization `kconfig`

Mail system maintenance

invokes the server for biff comsat
lists the contents of the mail queue mailq
rebuilds the database for the mail aliases file newaliases
sends mail sendmail

Monitoring system activity

displays load average statistics lav
displays the system status on the status line of a terminal sysline
gathers printer/plotter accounting information pac
generates a system activity graph sag
generates disk accounting data by user ID diskusg
invoke connect-time accounting acctcon
invoke connect-time accounting acctcon1
invoke connect-time accounting acctcon2
logs system messages syslogd
manipulate connect accounting records fwtmp
manipulate connect accounting records wtmpfix
merges or adds accounting files acctmerg
present an overview of accounting commands acct
provide process accounting acctprc
provide process accounting acctprc1
provide process accounting acctprc2
provide shell procedures for accounting acctsh
provide shell procedures for accounting chargefee
provide shell procedures for accounting ckpacct
provide shell procedures for accounting dodisk
provide shell procedures for accounting lastlogin
provide shell procedures for accounting monacct
provide shell procedures for accounting nulladm
provide shell procedures for accounting prctmp
provide shell procedures for accounting prdaily
provide shell procedures for accounting prtacct

provide shell procedures for accounting shutacct
 provide shell procedures for accounting turnacct
 report system activity sa1
 report system activity sa2
 report system activity sadc
 reports interprocess communication facilities status ipcs
 reports system activity sar
 runs daily accounting runacct
 runs startup programs at boot time startup
 searches and formats process accounting files acctcom
 summarizes commands from per-process accounting records acctcms

Network File System (NFS) network maintenance

displays Network File System (NFS) statistics nfsstat
 exports and unexports directories to Network File System
 (NFS) clients exportfs
 handle local and remote lock requests lockd
 invoke the NFS daemons biod
 invoke the NFS daemons nfsd
 invokes the Network File System (NFS) mount-request server mountd
 mounts Network File System (NFS) when needed automount
 provide crash and recovery monitoring for network locking
 services statd
 reports RPC information rpcinfo
 returns information for the `spray` command sprayd
 sets or displays the name of the Network Information
 Service (NIS) domain domainname
 shows all remote mounts showmount
 sprays packets spray

Print spooler maintenance

allows lp requests accept
configures the lp spooling system lpadmin
controls the operation of the line printer lpc
enable or disable LP printers disable
enable or disable LP printers enable
filter data for the POSTSCRIPT printers psbanner
filter data for the POSTSCRIPT printers pscomm
filter data for the POSTSCRIPT printers psinterface
filter data for the POSTSCRIPT printers psrv
filter data for the POSTSCRIPT printers pstext
filter data for the POSTSCRIPT printers transcript
generates a line-printer ripple pattern lptest
prevents LP requests reject
prints lp status information lpstat
start or stop the lp request scheduler and move requests lpmove
start or stop the lp request scheduler and move requests lpsched
start or stop the lp request scheduler and move requests lpshut
supports the Berkeley print spooler lpd

Setting up the system

adds disk blocks to or deletes them from the swap area swap
associates named partitions with device files pname
changes the current A/UX system node name chgnod
checks the installation of boards checkinstall
compiles (translates) terminfo files tic
compiles time-zone information files that are required to set the
local time-zone tzic
displays the date and time for one or more time zones tzdump
formats a disk through a driver-dependent format
operation diskformat
modify the /etc/inittab file in terms of enabling serial
ports for use as login terminals tty_add
modify the /etc/inittab file in terms of enabling
serial ports for use as login terminals tty_kill

performs disk partitioning dp
 pushes streams line disciplines line_sane
 set or reset the terminal to a sensible state reset
 set or reset the terminal to a sensible state tset
 set the initial communication modes, such as speed and
 line discipline, for the purpose of logging users
 in to A/UX through serial lines apm_getty
 set the initial communication modes, such as speed and line
 discipline, for the purpose of logging users in to A/UX
 through serial lines getty
 sets or displays the identifier of the current host system hostid
 sets or displays the name of the current host system hostname
 sets or updates bad block information badblk
 sets the characteristics of a serial port setport
 sets the keyboard for the console keyset
 sets the local time zone settimezone

Starting up and shutting down

displays a progress bar during the A/UX boot
 sequence StartMonitor
 execute system initialization shell scripts bcheckrc
 execute system initialization shell scripts brc
 execute system initialization shell scripts macsysinitrc
 execute system initialization shell scripts powerfail
 execute system initialization shell scripts rc
 execute system initialization shell scripts sysinitrc
 kills all active processes killall
 reboots the operating system reboot
 runs startup programs at boot time startup
 sends messages to StartMonitor, which displays a
 progress bar during the A/UX boot process
 startmsg
 spawn general processes init
 spawn general processes telinit
 terminates processes that support multi-user mode and
 enters single-user mode shutdown

turns off power to the computer powerdown

System administration tools

builds a device file mknod
removes device files from a directory dev_kill
substitutes user ID su

TCP/IP network maintenance

attaches a serial line to a network interface and configures
the network interface slattconf
attaches a serial line to a network interface slattach
converts Internet addresses to standard form stdhosts
converts RPC program numbers into DARPA protocol port
numbers portmap
creates or updates the Compressed Serial Line/Internet
Protocol (CSL/IP) database mkslipuser
displays and modifies the address translation table arp
displays network status information netstat
displays the current state of the Compressed Serial
Line/Internet Protocol (CSL/IP) database
..... dslipuser
displays the Ethernet address of each Ethernet card in
your system etheraddr
exercises the TCP/IP network by sending Internet Control
Message Protocol (ICMP) packets to a named host
..... ping
interactively queries name servers nslookup
invokes a server for kernel statistics rstatd
invokes the network routing daemon routed
invokes the network rwall server rwalld
invokes the remote shell server remshd
invokes the remote user communication server talkd
invokes the system status server rwhod
manages network interfaces ifconfig
manipulates the routing tables route
prints a readable description of TCP trace records trpt

provide Internet File Transfer Protocol (FTP) service ftpd
 provides Internet domain name service named
 runs on a remote system to log you in remlogin
 server for remote executions rexecd
 server for remote logins rlogind
 starts Internet servers when needed inetd
 supports the DARPA standard TELNET protocol telnetd
 rusers invokes a server for users rusersd

User account maintenance

adds a user account adduser
 changes the real-name field of your password file entry for use by
 finger chfn
 check the password/group files grpck
 check the password/group files pwck
 edits the password file vipw
 handles requests from remote systems for user information
 from finger fingerd

UUCP network maintenance

checks the uucp directories and files uucheck
 cleans up files in the uucp spool directory uudemond.cleanup
 contacts a remote system with debugging on Uutry
 handles remote mail received via UUCP rmail
 handles requests from remote systems to run commands uuxqt
 handles the transfer of files by uucico over TCP/IP
 connections uucpd
 mails current uucp work status to the uucp
 administrator uudemond.admin
 processes spooled uucp requests uudemond.hour
 removes old files from the uucp spool directory uucleanup
 schedules uucp file transfers uusched
 sets up polling for selected systems uudemond.poll
 transfers files as specified by uucp work files uucico

Name Information Server (NIS) maintenance

changes a login password on the Network Information
Service (NIS) master server yppasswd

displays the host name of a system's Network Information
Service (NIS) server ypwhich

generates a Network Information Service (NIS) dbm file makedbm

handle requests to change a password served by the
Network Information Service (NIS) yppasswdd

initializes Network Information Service (NIS) maps for
master and slave servers ypinit

lists the contents of a Network Information Service (NIS) map ypcat

lists the value of a specified key in a Network Information
Service (NIS) map ypmatch

propagates changed Network Information Service (NIS) maps yppush

provide Network Information Service (NIS) service ypserv

rebuilds the Network Information Service (NIS) maps ypmake

reports the version of a Network Information Service (NIS)
map that is on an NIS server yppoll

reverses the netgroup file revnetgroup

sets ypbind to a particular domain and Network Information
Service (NIS) server ypset

transfers a Network Information Service (NIS) map to the local
system ypxfr

Command Synopses

300

300 [+12] [-half-line-units] [-dtab-delay, line-delay, char-delay]
300s [+12] [-half-line-units] [-dtab-delay, line-delay, char-delay]

300s

See 300.

4014

4014 [-ccolumns] [-n] [-plines[i]] [l [-t]] [file]

450

450

accept

accept *destinations*

acct

acctdisk
acctdusg [-p *file*] [-u *file*]
accton [*file*]
acctwtmp *reason*

acctcms

/usr/lib/acct/acctcms [-a [-o] [-p]] [-c] [-j] [-n] [-s] [-t]
file...

acctcom

acctcom [-a] [-b] [-C *sec*] [-e *time*] [-E *time*] [-f] [-g *group*]
[-h] [-H *factor*] [-i] [-I *chars*] [-k] [-l *line*] [-m] [-n *pattern*]
[-o *ofile*] [-O *sec*] [-q] [-r] [-s *time*] [-S *time*] [-t] [-u *user*] [-v]
[*file*]...

acctcon

acctcon1 [-l*file*] [-o*file*] [-p] [-t]
acctcon2

acctcon1

See acctcon.

acctcon2

See acctcon.

acctdisk

See acct.

acctdusg

See acct.

acctmerg

acctmerg [-a] [-i] [-p] [-t] [-u] [-v] *file*...

accton

See acct.

acctprc

acctprc1 [ctmp]

acctprc2

acctprc1

See acctprc.

acctprc2

See acctprc.

acctsh

chargefee *login-name number*

ckpacct [*amt*]

dodisk [-o] [*filesys*]...

lastlogin

monacct *month*

nulladm *name*

prctmp [*recfile*]...

prdaily [-l] [-c] [*mdd*]

prtacct *file [heading]*

shutacct [*reason*]

startup

turnacct on| off| switch

acctwtmp

See acct.

adb

adb [-k] [-w] [*object-file* [*core-file*]]

addbib

addbib [-a] [-p *prompt-file*] *database*

adduser

adduser [-a *address*] [-c] [-d *dir*] [-g *group*] [-h *home*] [-i] [-p *home-phone*] [-r *real-name*] [-s *shell*] [-u *lowest*] [-U *uid*] [-x *extension*] [*login-name*]...

admin

admin [-a*name-or-gid*] [-d*option*[*value*]] [-e*name-or-gid*] [-f*option*[*value*]] [-h] [-i[*name*]] [-m[*mlist*]] [-n] [-r*release*[*level*]] [-t[*descriptive-text*]] [-y[*comment*]] [-z] *file*...

ae**aliens**

aliens

apm_getty

See getty.

appleping

appleping *net-node* [*packet-size* [*npackets*]]
appleping *name* : *type*[@*zone*] [*packet-size* [*npackets*]]

appletalk

appletalk [-b *hardware-interface*] [-c] [-d] [-i *interface*] [-n] [-p] [-s] [-u] [-z]

apply

apply [-a*esc-char*] [-a*rgs-per-batch*] *command* *argument*...

apropos

apropos *search-string*...

ar

```

ar -dp [l] [v] archive file...
ar -mp [l] [v] [position archivefile] archive file...
ar -qp [c] [l] [v] archive file...
ar -rp [c] [l] [u] [v] [position archivefile] archive file...
ar -tp [s] [v] archive file...
ar -xp [l] [s] [v] archive file...

```

arithmetic

```

arithmetic [+ ] [- ] [x] [/] [range]

```

arp

```

arp host
arp -a [kernel] [memory-interface]
arp -d host
arp -f file
arp -s host ethernet-address [status]

```

arp**as**

```

as [-A factor] [ -m ] [ -n ] [-o object-file] [ -R ] [ -V ]
[ -68030 ] [ -68040 ] [ -68851 ] file

```

asa

```

asa [file]...

```

ascii

```

cat /usr/pub/ascii

```

at

```

at time [day] [+ increment]
at -l [job-number]...
at -r job-number...
batch

```

atlookup

```

atlookup [-d] [-r nn] [-s ss] [-x] [object[:type[@zone]]]
atlookup -z [-C]

```

atprint

```

atprint [printer-name[:printer-type[@zone]]]

```

atstatus

atstatus [*object* [:*type* [@*zone*]]]

at_cho_prn

at_cho_prn [*type*[@*zone*]]

autoconfig

autoconfig [-a] [-b *module-directory*] [-d *init-scripts-directory*]
 [-i *base-kernel*] [-I] [-k] [-l *linker*] [-L *loadfile*]
 [-m *master-directory*] [-M *master-file*] [-o *kernel-file*]
 [-s *startup-scripts-directory*] [-S *script*] [-t *timeout*] [-v] [-V]
 autoconfig -D [-i *base-kernel*] [-v] [-V]
 autoconfig -c [-v] [-V]

automount

automount [-D *environment-variable=value*] [-f *master-file*] [-m]
 [-M *mount-directory*] [-n] [-tl *duration*] [-tm *interval*]
 [-tw *interval*] [-T] [-v] [*directory map* [-*mount-options*]]...

autorobots

autorobots

awk

awk [-F*field-separator*] '*pattern-action...*' [[-v] *variable=value*]...
 [*file*]...
 awk [-f *awk-source-file*] [-F*field-separator*] [[-v] *variable=value*]...
 [*file*]...

back

back

badblk

badblk [-r] /dev/rdsk/c?d?s? [*blkno*]...

banner

banner *string*...

banner7

banner7 [-w [*width*]] [*text*]

basename

basename *string* [*suffix*]
 dirname *string*

batch

See at.

bc

bc [-c] [-l] [*file*]...

bcd

bcd *text*

bcheckrc

See brc.

bcopy

bcopy

bdiff

bdiff *file1 file2* [*lines-per-segment*] [-s]

bfs

bfs [-] *file*

biff

biff [*switch*]

biod

See nfsd.

bj

bj

brc

brc
bcheckrc
macsysinitrc
powerfail
rc
sysinitrc

bs

bs [*file* [*argument*]...]

cal

cal [[*month*] *year*]

calendar

calendar [-]

cancel

cancel [*printer*]
cancel [*id*]...

cat

cat [-] [-e] [-s] [-t] [-u] [-v] [*file*]...

cb

cb [-j] [-l *line-length*] [*file*]...
cb [-j] [-s] [*file*]...

cc

cc [-A *factor*] [-a] [-B *string*] [-c] [-C] [-Dsymbol[=*def*]] [-E]
[-fm68881] [-F] [-g] [-I*dir*] [-lx] [-L *dir*] [-n] [-o *outfile*] [-O]
[-p] [-P] [-R] [-s] [-S] [-t [*p012al*]] [-T] [-Usymbol] [-v]
[-W *c, arg1[, arg2]*]... [-X] [-y] [-Z*flags*] [-68030] [-68040]
[-68851] [-#]... *file*...

ccat

See compact.

cdc

cdc [-m[*mrlist*]] -r *SID* [-y[*comment*]] *file*...

cflow

cflow [-dnum] [-i_] [-ix] [-r] *file*...

changesize

changesize [\pm *option*] [-mminsize] [-pprefsize] [-v] *file*

chargefee

See acctsh.

chase

chase [*nrobots*] [*nfences*]

checkcw

See cw.

checkeq

See eqn.

checkinstall

checkinstall ethertalk

checkmm

checkmm *file...*

checkmm1

See checkmm.

checknr

checknr [-a .x1 .y1 .x2 .y2xn .yn] [-c .x1 .x2 .x3xn] [-f]
[-s] [*file*]...

chfn

chfn [*login-name*]

chgnod

chgnod *new-nodename* [*kernel-file*]

chgrp

See chown.

chmod

chmod *mode file...*

chown

chown *owner file...*

chgrp *group file...*

chroot

chroot *newroot command*

chsh

chsh *name* [*shell*]

ci

ci [-f[*rev*]] [-k[*rev*]] [-l[*rev*]] [-q[*rev*]] [-r[*rev*]] [-u[*rev*]] [-mmsg]
[-nname] [-Nname] [-sstate] [-t[*txtfile*]] *files*

ckpacct

See acctsh.

clear

clear

clri

clri [-T*file-system-type*] *file-system* *i-number*...

cmdo

cmdo *command*

cmdo -o *resfile* [-n] [-s] *command*

cmp

cmp [-l] [-s] *file1* *file2*

co

co [-ddate] [-jjoinlist] [-l[rev]] [-p[rev]] [-q[rev]] [-r[rev]] [-sstate]
[-w[login]] *files*

col

col [-b] [-f] [-p] [-x]

colcrt

colcrt [-] [-2] [*file*]

colrm

colrm *startcol* [*endcol*]

comb

comb [-clist] [-o] [-psid] [-s] *file*...

comm

comm [- [1] [2] [3]] *file1* *file2*

CommandShell

CommandShell [-b *macsysinit-pid*] [-q] [-u]

compact

compact [*name*]...

uncompact [*name*]...

ccat [*file*]...

compress

```
compress [-b maxbits] [-c] [-f] [-v] [-V] [file]...
compressdir [compress-flag]... [directory]...
uncompressdir [uncompress-flag]... [directory]...
uncompress [-c] [-f] [-v] [-V] [file]...
zcat [-V] [file]...
zcmp [cmp-option]... file1 [file2]
zdiff [diff-option]... file1 [file2]
zmore [file]...
```

compressdir

See compress.

comsat

comsat

conv

```
conv [-] [-a] [-o] [-p] [-s] -ttarget file...
```

cp

```
cp [-i] [-r] file1 file2
cp [-i] [-r] file... directory
```

cpio

```
cpio -o [a] [c] [B] [F] [v]
cpio -i [6] [b] [B] [c] [d] [f] [m] [r] [s] [S] [t] [u] [v] [patterns]
cpio -p [a] [d] [l] [m] [u] [v] directory
```

cpp

```
cpp [-C] [-Dname[=def]] [-Idir] [-P] [-Uname] [-M[prefix]] [-Y]
[ifile [ofile]]
```

cpset

```
cpset [-o] [-O] file directory [mode [owner [group]]]
```

craps

craps

cribbage

```
cribbage [-e] [-q] [-r] name...
```

cron

cron

crontab

```
crontab [file]
crontab -l
crontab -r
```

crypt

```
crypt [password]
```

csch

```
csch [-c] [-e] [-f] [-i] [-n] [-s] [-t] [-v] [-V] [-x] [-X] [arg]...
```

csplit

```
csplit [-f prefix] [-k] [-s] file arg1 [... argn]
```

ct

```
ct [-cdevice-type] [-h] [-ldevice-name] [-sbaud-rate] [-v]
[-wtime-limit] [-xdebug-level] telephone-number ...
```

ctags

```
ctags [-a] [-u] [-w] [-x] file...
```

ctrace

```
ctrace [-b] [-e] [-ffunctions] [-ln] [-o] [-p 's'] [-P] [-rf] [-s]
[-tn] [-u] [-vfunctions] [-x] [file]
```

cu

```
cu [-bbits] [-dhint] [-e] [-cdevice-type] [-o] [-sbaud-rate]
[-xdebug-level] -ldevice-name
cu [-bbits] [-dhint] [-e] [-cdevice-type] [-ldevice-name] [-o]
[-sbaud-rate] [-xdebug-level] telephone-number
cu [-bbits] [-dhint] [-e] [-cdevice-type] [-ldevice-name] [-o]
[-sbaud-rate] [-xdebug-level] system
```

cubic

See ttt.

cut

```
cut -clist [-s] [file]...
cut -flist [-d char] [-s] [file]...
```

cw

```
cw [-d] [-fn] [-lxx] [-rxx] [-t] [+t] [file]...
checkcw [-lxx] [-rxx] file...
```

cxref

cxref [-c] [-o *file*] [-s] [-t] [-w[*num*]] *file*...

daps

daps [-b] [-hstring] [-olist] [-r] [-sn] [-t] [-w] [*file*]...

date

date [*mmddhhmm*[*yy*]] [+*format*]

dbx

dbx [-c *file*] [-D] [-i] [-I *dir*]... [-r] [*objfile* [*coredump*]]

dc

dc [*file*]

dcopy

dcopy [-an] [-d] [-fsize [:*isize*]] [-sX] [-v] *inputfs outputfs*

dd

dd [bs=*n*] [cbs=*n*] [conv=ascii] [conv=ebcdic] [conv=ibm]
 [conv=lcase] [conv=noerror] [conv=swab] [conv=sync]
 [conv=*type*, *type*] [conv=ucase] [count=*n*] [ibs=*n*] [if=*file*]
 [multi=in] [multi=in, out] [multi=out] [of=*file*] [obs=*n*]
 [seek=*n*] [skip=*n*]

delta

delta [-glist] [-m[*mrlist*]] [-n] [-p] [-rSID] [-s] [-y[*comment*]] *file*...

derez

derez [-c] [-dmacro-assignment]... [-e] [-iinclude-dir]...
 [-mstring-size] [-p] [-rd] [-umacro]... *resource-file*
 [*resource-description-file*]...
 derez [-c] -oscope [-dmacro-assignment]... [-e] [-iinclude-dir]...
 [-mstring-size] [-p] [-rd] [-umacro]... *resource-file*
 [*resource-description-file*]...
 derez [-c] -somit-scope [-dmacro-assignment]... [-e]
 [-iinclude-dir]... [-mstring-size] [-p] [-rd] [-umacro]... *resource-file*
 [*resource-description-file*]...

deroff

deroff [-mx] [-w] [*file*]...

devnm

devnm [*mount-point*]

dev_kill

dev_kill *number directory...*

df

df -t [-f] [-T *fs-type*] [*fs-reference*]...

df -B [-i] [-T *fs-type*] [*fs-reference*]...

df -p [-i] [-T *fs-type*] [*fs-reference*]...

diction

diction [-f *pfile*] [-ml] [-mm] *file...*

diction [-ml] [-mm] [-n] *file...*

explain

diff

diff [-b] [-c] [-e] [-f] [-h] [-l] [-r] [-s] [-Sname] *dir1 dir2*

diff [-b] [-c] [-e] [-f] [-h] *file1 file2*

diff [-b] *file1 file2*

diff3

diff3 [-3] [-e] [-x] *file1 file2 file3*

diffmk

diffmk [-] *file1 file2 file3*

dircmp

dircmp [-d] [-s] [-wn] *dir1 dir2*

dirname

See *basename*.

dis

dis [-d *sec*] [-da *sec*] [-F *function*] [-l *string*] [-L] [-o] [-t *sec*]

[-V] *file...*

disable

See *enable*.

diskformat

diskformat [-cyl *start*[-*end*]] [-dens *n*] [-head 0] *floppy-device*
 diskformat [-size 532] *hard-disk-device*

diskusg

diskusg [-i *ignlist*] [-p *pw-file*] [-s] [-u *outfile*] [-v] [*file*]...

dodisk

See acctsh.

domainname

domainname [*domain-name*]

dp

dp [-q] [-u] *file*

dslipuser

dslipuser

du

du [-a[1]] [-r] [-s] [*files*]

dump

dump [[-a] [-c] [-f] [-g] [-h] [-l] [-o] [-r] [-s] [-t] [-z *name*]]
 [[-d *number*] [+d *number*] [-n *name*] [-p] [-t *index*] [+t *index*]
 [-u] [-z *name, number*] [+z *name*]] *file*...
 dump [[-a] [-c] [-f] [-g] [-h] [-l] [-r] [-t] [-z *name*]]
 [[-d *number*] [+d *number*] [-n *name*] [-p] [-t *index*] [+t *index*]
 [-u] [-v] [-z *name, number*] [+z *name*]] *file*...

dump.bsd

dump.bsd [Tfstype] [dumplev] [b] [d] [f] [n] [s] [u] [bdfs-arg]...
fs-reference

dump.bsd [Tfstype] [dumplev] c [d] [f] [n] [s] [u] [dfs-arg]... *fs-reference*

dump.bsd [Tfstype] [dumplev] F [d] [f] [n] [s] [u] [dfs-arg]... *floppydev*

dump.bsd [Tfstype] w

dump.bsd [Tfstype] W

rdump [Tfstype] [dumplev] [b] [d] [f] [n] [s] [u] [bdfs-arg]...
host:fs-reference

rdump [Tfstype] [dumplev] c [d] [f] [n] [s] [u] [dfs-arg]... *host:fs-reference*

rdump [Tfstype] [dumplev] F [d] [f] [n] [s] [u] [dfs-arg]... *host:fs-reference*

rdump [Tfstype] w

rdump [Tfstype] W

e

See ex.

echo

echo [arg]...

ed

ed [-] [-p *string*] [-x] [*file*]
red [-] [-p *string*] [-x] [*file*]

edit

See ex.

efl

efl [-#] [-C] [-w] [*file*]...

egrep

See grep.

eject

eject [0] [1] [/dev/rdisk/*name*]

enable

enable *printers*
 disable [-c] [-r[*reason*]] *printers*

enscript

enscript [-1] [-2] [-bheader] [-B] [-ffont] [-Fhfont] [-g] [-G] [-h]
 [-k] [-K] [-l] [-Llines] [-m] [-o] [-pout] [-q] [-r] [-R] [[-#n]
 [-Cclass] [-Jname] [-Pprinter]] [*files*]
 enscript [-1] [-2] [-bheader] [-B] [-ffont] [-Fhfont] [-g] [-G] [-h]
 [-k] [-K] [-l] [-Llines] [-m] [-o] [-pout] [-q] [-r] [-R] [[-ddest]
 [-nn] [-ttitle] [-w] [*files*]

env

env [-] [*name=value*]... [*command args*]

environ

extern char **environ;

eqn

eqn [-dxy] [-fn] [-pn] [-sn] [-Ttty-type] [-] [*file*]...
 checkeq [*file*]...

eqnchar

eqn /usr/pub/eqnchar [*options*] [-] [*files*] | troff [*options*]
 eqn /usr/pub/cateqnchar [*options*] [-] [*files*] | troff [*options*]
 neqn /usr/pub/eqnchar [*options*] [-] [*files*] | troff [*options*]
 eqn -Taps /usr/pub/apseqnchar [*options*] [-] [*files*] | troff
 [*options*]

errdead

/etc/errdead *dumpfile* [*namelist*]

errdemon

errdemon [*file*]

errpt

errpt [-dev] [-a] [-e *date*] [-f] [-p *n*] [-s *date*] [*file*]...

errstop

errstop [*namelist*]

esch

esch [-b] [-c *cluster-number*] [-f] [-v]

escher

escher [-y] [-m]
escher *file*...

etheraddr

etheraddr [*slot*]

eu

eu *file*

eupdate

eupdate

ex

ex [-] [+*command*] [-r] [-R] [-t *tag*] [-v] [-x] *file*...
e [-] [+*command*] [-r] [-R] [-t *tag*] [-v] [-x] *file*...
edit [-] [+*command*] [-r] [-R] [-t *tag*] [-v] [-x] *file*...

expand

expand -a [-*tabstop*] [-*tab1*, *tab2*, ... , *tabn*] [*file*]...
unexpand [*file*]...

explain

See diction.

exportfs

exportfs
exportfs -a [-i] [-v] [-o *export-options*] [-u] [*directory-or-file*]...
exportfs -u [-v] *directory-or-file*...

expr

expr *arguments*

exterr

exterr /dev/*devicename* [*choice*]

f77

f77 [-1] [-66] [-A *factor*] [-c] [-C] [-E] [-f] [-F] [-g] [-I[*24s*]]
[-m] [-N*tableentries*]... [-o*output*] [-O] [-onetrip] [-p] [-R] [-S]
[-u] [-U] [-w] *file*...

factor

factor [*number*]

false

See true.

fcntl

#include <fcntl.h>

fcnv

fcnv [-f] [-v] [-i *start-format*] -s *input-file* *output-file*
 fcvt [-f] [-v] [-i *start-format*] -d *input-file* *output-file*
 fcvt [-f] [-v] [-i *start-format*] -t *input-file* *output-file*
 fcvt [-f] [-v] [-i *start-format*] -p *input-file* *output-file*
 fcvt [-f] [-v] [-i *start-format*] -b *input-file* *output-file*
 fcvt [-f] [-v] [-i *start-format*] -m *input-file* *output-file*

ff

ff [-an] [-cn] [-i *inode-list*] [-I] [-l] [-mn] [-n *file*] [-pprefix] [-s]
 [-u] *device-file*

fgrep

See grep.

file

file [-c] [-f *ffile*] [-m *mfile*] *arg...*

finc

finc [-a *n*] [-c *n*] [-m *n*] [-n *file*] *disk-device-file* *device-file*

find

find *pathname...* *expression*

finger

finger [f] [w] [*login-or-real-name*]...
 finger -i [f] [w] [*login-name*]...
 finger -q [f] [w] [*login-name*]...
 finger -l [b] [h] [m] [p] [*login-or-real-name*]...
 finger [-l] *login-or-real-name@host* [*login-or-real-name@host*]...
 finger [-s] *@host* [*@host*]...

fingerd

in.fingerd

finstall

finstall

fish

fish

fmtfmt [*file*]...**fold**fold [-width] [*file*]...**font**troff -T*ty-type* ...**fortune**

fortune

fpr

fpr

freqfreq [-freqfile *inumber:name...*] [-ppath] *device-file***freq**freq [*file*]...**from**from [-s *sender*] [*user*]**fsck**

fsck -T*fs-type* [-y] [-n] [-m *timeout*] [-s *interleave*] [-S *interleave*]
[-t *file*] [-q] [-D *option*]... [-f] [-p *passtostart*] [*syfs-filesystem*]...
fsck -T*fs-type* [-b *block-number*] [-y] [-n] [-m *timeout*]
[-p *passtostart*] [*ufs-filesystem*]...

fsdb

fsdb [-?] [-o] [-pstring] [-T4.2] [-w] *UFS-symbol*...
fsdb [-] [-T5.2] *SVFS-symbol*...

fsentry

fsentry -t *type* [-o *optlist*] [-d *dumpfreq*] [-p *passno*] [-n] [-f]
disk-device-file mount-point

fsirand

fsirand [-p] [-T*fs-type*] *special*

fsplit

fsplit [-e] [-f] [-s] *file...*

fsstat

fsstat [-T*fs-type*] *fs-device-file*

fstyp

fstyp *file*

ftp

ftp [-d] [-g] [-i] [-n] [-v] [*remote-system*]

ftpd

ftpd [-d] [-l] [-t*timeout*]

fuser

fuser [-] [-k] [-n*namelist*] [-u] *file...*

fwtmp

fwtmp [-ic]
wtmpfix [*file*]...

get

get [-a*seq-no*] [-b] [-c*cutoff*] [-e] [-g] [-i*list*] [-k] [-l[*p*]] [-m] [-n]
[-p] [-r*SID*] [-s] [-t] [-w*string*] [-x*list*] *file...*

getopt

getopt [*flag-letter[:]*]... [*input-string*]

getty

getty [-C *string*] [-d] [-h] [-i] [-q] [-t *timeout*] *line*
[*gettydefs-label* [*type* [*linedisc*]]]
getty -c *file*

apm_getty [-h] [-t *timeout*] *line* [*gettydefs-label* [*type* [*linedisc*]]]

grap

grap [-T*tty-type*] [-l] [-] [*file*]...

graph

graph [-a *sp*] [*st*] [-b] [-c*label*] [-g *style*] [-h *hspace*] [-l *title*]
 [-m*mode*] [-r *rspace*] [-s] [-t] [-u *uspace*] [-w *wspace*] [-x [l] [a]
 [b] [c]] [-y [l] [a] [b] [c]]

greek

greek [-T*terminal*]

greek

greek -T*terminal*[</usr/pub/greek]

grep

grep [-b] [-c] [-i] [-l] [-n] [-s] [-v] *expression* [*file*]...
 egrep [-b] [-c] [-e *expression*] [-f *file*] [-i] [-l] [-n] [-v]
 [*expression*] [*file*]...
 fgrep [-b] [-c] [-e *expression*] [-f *file*] [-i] [-l] [-n] [-v] [-x]
 [*strings*] [*file*]...

groups

groups [*user*]

grpck

See pwck.

hangman

hangman [*dictionary*]

hashcheck

See spell.

hashmake

See spell.

head

head [-*count*] [*file*]...

help

help [*args*]...

hex

hex [-f] [-l] [-n#] [-ns8] [-r] [-s0] [-s2] [+*saddr*] *ifile*

hostid

hostid [*identifier*]

hostname

hostname [*nameofhost*]

hyphen

hyphen [*file*]...

icmp

None; included automatically with inet(5F).

id

id

ident

ident *file*...

ifconfig

ifconfig *interface* [*address[dest-address]*] [*option*]...

ifconfig *interface* [*address-family*]

in.fingerd

See fingerd.

in.ftpd

See ftpd.

in.tftpd

See tftpd.

indent

indent *input* [*output*] [-bc, -nbc] [-br, -bl] [-cn] [-cdn]
[[-dj], -ndj] [-dn] [-in] [-ln] [-v, -nv]

indxbib

indxbib [*database*]... [*file*]...

inet

```
#include <sys/types.h>
```

```
#include <netinet/in.h>
```


inetd

inetd [-d]

init

init [*run-level* [*directive*]]

install

install [-c *dira*] [-f *dirb*] [-g *group*] [-i] [-m *mode*] [-n *dirc*]
[-o] [-s] [-u *user*] *file* [*dirx*]...

install [-c *dira*] [-s] *file* [*dirx*]...

install [-f *dirb*] [-o] [-s] *file* [*dirx*]...

install [-g *group*] [-i] [-m *mode*] [-n *dirc*] [-o] [-s] [-u *user*]
file [*dirx*]...

ip

#include <sys/socket.h>

#include <netinet/in.h>

ipcrm

ipcrm [-m *shmid*] [-M *shmkey*] [-q *msqid*] [-Q *msgkey*] [-s *semid*]
[-S *semkey*]

ipcs

ipcs [-a] [-b] [-c] [-C *corefile*] [-m] [-N *namelist*] [-o] [-p] [-q]
[-s] [-t]

isotomac

See mactoiso.

iw2

iw2 [-a *dotSPACE*] [-b] [-c *color*] [-d] [-D *udcfile*] [-f] [-h]
[-k *mode*] [-l *language*] [-m *margin*] [-n *length*] [-o *file*]
[-p *pitch*] [-q *quality*] [-s *spacing*] [-t *tabs*] [-u] [-U *udcfile*]
[-w *value*] [-x] [-z] [*file*]...

join

join [-an] [-e *string*] [-jn *m*] [-o *list*] [-tc] *file1 file2*

kconfig

kconfig [-a [-v] [-V]] [-n*namelist*]

kermit

kermit [-a *fnl*] [-b *n*] [-c] [-d] [-f] [-g *rfn*] [-h] [-i] [-k]
[-l *dev* [-n]] [-p *x*] [-q] [-r] [-s *fn*] [-t] [-w] [-x] [*file*]...

keyset

keyset [-c *country*] [-k *keyboard*]

kill

kill [-sig] *pid*...

killall

killall [-n *namelist*] [*signal*]

ksh

ksh [-a] [-c *string*] [-e] [-f] [-h] [-i] [-k] [-m] [-n] [-o *option*]...
[-p] [--*positional-arg*]... [*±positional-arg*]... [-r] [-s] [-t] [-u] [-v]
[-x] [*file*]...

labelit

See volcopy.

last

last [*name*]... [*tty*]...

lastlogin

See acctsh.

launch

launch [-adr] *application* [*document*]...
launch -p [*adr*] *application* *document*...

launch

launch -a [-d] [-e] [-f] [-k *value*] [-m]
[-p *swapdev-spec*] [-r] [-v] [-S] [-s] [*path*]

launch -n [-d] [-e] [-f] [-k *value*] [-m]
[-p *swapdev-spec*] [-r] [-v] [-s] [-S] [*path*]

lav

lav

ld

ld [-afactor] [-e *epsym*] [-f *fill*] [-ild] [-lx] [-m] [-o *outfile*] [-r] [-s] [-t] [-u *symname*] [-x] [-z] [-F] [-Ldir] [-M] [-N] [-S] [-V] [-VS *num*] *file*...

leave

leave [*hhmm*]

lex

lex [-c] [-n] [-t] [-v] [*file*]...

life

life [-r]

line

line *input*

line_sane

line_sane [*filde*s]

lint

lint [-a] [-b] [-Dname[=*def*]] [-h] [-Idir] [-lx] [-n] [-o *lib*] [-p] [-u] [-Uname] [-v] [-x] *file*...

ln

ln [-s] *file1* [*file2*]
ln *file*... *directory*
ln -f *directory1* *directory2*

lo

pseudo-device loop

lockd

See `rpc.lockd`.

login

login [*name* [*env-var*...]]

Login

Login [*startmac-options*] [-- [-g] [-r]]

logname

logname

lookbib

lookbib [-n] *database*

lorder

lorder *file...*

lp

lp [-c] [-d*dest*] [-m] [-n*number*] [-o*option*] [-s] [-t*title*] [-w] [*file*]...

lpadmin

lpadmin -p*printer* [-c*class*] [-e*printer*] [-h] [-i*interface*] [-l]
 [-m*model*] [-r*class*] [-v*device*]
 lpadmin -x*dest*
 lpadmin -d [*dest*]

lpc

lpc [*command* [*argument*]...]

lpd

lpd [-l] [*alt-internet-no*]

lpmove

See lpsched.

lpq

lpq [+*sleep-interval*] [-l] [-P*printer*] [*jobno*]... [*user*]...

lpr

lpr [-# *copies*] [-C *class*] [-h] [-i [*indent-cols*]] [-J *cover-title*] [-l]
 [-m] [-p] [-P *printer*] [-r] [-s] [-T *title*] [-w*page-width*] [*file*]...

lprm

lprm [-P*printer*] [-] [*jobno*]... [*user*]...

lpsched

lpsched
 lpshut
 lpmove *requests dest*
 lpmove *dest1 dest2*

lpshut

See lpsched.

lpstat

lpstat [-a $[list]$] [-c $[list]$] [-d] [-o $[list]$] [-p $[list]$] [-r] [-s] [-t]
[-u $[list]$] [-v $[list]$]

lptest

lptest [$length$ [$count$]]

ls

ls [-a] [-b] [-c] [-C] [-d] [-F] [-g] [-i] [-l] [-L] [-m] [-n] [-o]
[-p] [-q] [-r] [-R] [-s] [-t] [-u] [-x] [$names$]

m4

m4 [-B int] [-e] [-Hint] [-s] [-S int] [-T int] [-D $name$ [= val]] [-U $name$]
[$file$]...

m68k

See machid.

machid

m68k
pdp11
u3b
u3b2
u3b5
u3b15
vax

macquery

macquery [-a] [-c] [-n] [-s] [-t $timeout$] $resource-file$ $alert-id$
[$parm1$... $parm4$]

macref

macref [-n] [-s] [-t] [--] $file$...

macsysinitrc

See brc.

mactois

mactois [-c $char$] [$file$]
isotomac [-c $char$] [$file$]

mail

mail [-e] [-f $file$] [-p] [-q] [-r] [-t] $address$...

mailq

mailq [-v]

mailx

mailx [-d] [-e] [-f *filename*] [-F] [-h *number*] [-H] [-i] [-n]
[-N] [-r *address*] [-s *subject*] [-u *user*] [-U] [*name*]...

make

make [-a] [-b] [-B] [-ddigits] [-e] [-f *description-file*] [-g] [-G] [-i]
[-k] [-K] [-M] [-n] [-p] [-P] [-q] [-r] [-s] [-t] [-u] [-V] [*target*]...

makedbm

makedbm [-d *yp-domain-name*] [-i *yp-input-file*]
[-m *yp-master-name*] [-o *yp-output-name*] *infile outfile*
makedbm [-u *dbmfilename*]

makedev

makedev *files*

makekey

makekey

man

man [-c] [-d] [-T*term*] [-w] [[*section*] *name*]...

man

nroff -man *files*
troff -man [-rs1] *files*

mastermind

mastermind

math

#include <math.h>

maze

maze

me

nroff -me [*nroff-options*]...
troff -me [*troff-options*]...

merge

merge [-p] *file1 file2 file3*

mesg

mesg [*choice*]

mkdir

mkdir *dirname...*

mkfs

mkfs *device-file blocks[: inodes] [gap modulus]*

mkfs *device-file proto-file [gap modulus]*

mkfs1b

mkfs1b *special blocks[: inodes] [m n]*

mkfs1b *special proto [m n]*

mklost+found

mklost+found

mknod

mknod *name type [major minor]*

mknod *name p*

mkshlib

mkshlib -s *specs* [-n] -t *target* [-h *host*]

mkslipuser

mkslipuser

mkstr

mkstr [-] *messagefile prefix file...*

mm

mm [-12] [-c] [-e] [-E] [-t] [-Tty-type] [*file*]...

mm

mm [*options*] [*files*]

nroff -mm [*options*] [*files*]

nroff -cm [*options*] [*files*]

mmt [*options*] [*files*]

troff -mm [*options*] [*files*]

mmt

mmt [-a] [-D*ddest*] [-e] [-g] [-p] [-t] [-T*tty-type*] [-z] [*file*]...
 mvt [-a] [-D*ddest*] [-e] [-g] [-p] [-t] [-T*tty-type*] [-z] [*file*]...

module_dump

module_dump *kernel-file*

monacct

See acctsh.

moo

moo

more

more [-c] [-d] [-f] [-l] [-n] [-s] [-u] [+*linenumber*] [*file*]...
 more [-c] [-d] [-f] [-l] [-n] [-s] [-u] [+/*pattern*] [*file*]...
 page [-c] [-d] [-f] [-l] [-n] [-s] [-u] [+*linenumber*] [*file*]...
 page [-c] [-d] [-f] [-l] [-n] [-s] [-u] [+/*pattern*] [*file*]...

mount

mount [-p]
 mount -a [f] [r] [v] [-t *type*] [-T *type*]
 mount [-f] [r] [v] [-t *type*] [-T *type*] [-o *options*] *device-file*
mount-point
 umount [-v] -h *host*
 umount -a [v]
 umount [-v] [*device-file*]...
 umount [-v] [*mount-point*]...

mountd

rpc.mountd [-n]

mptx

nroff -mptx [*options*] [*files*]
 troff -mptx [*options*] [*files*]

ms

nroff -ms [*nroff-options*]...
 troff -ms [*troff-options*]...

mt

mt [-f*device-file*] *command* [*count*]

mv

```
mv [-i] [-f] [-] file1 file2
mv [-i] [-f] [-] file... directory
```

mv

```
mvt [-a] [options] [files]
troff [-a] [-rX1] -mv [options] [files]
```

mvt

See mmt.

named

```
named [-d debuglevel] [-p port#] [bootfile]
```

ncheck

```
ncheck [-a] [-i i-node-numbers] [-s] [-Tfile-system-type]
[file-system]
```

ncstats

```
ncstats
```

ndx

```
ndx subjfile formatter-command-line
```

neqn

```
neqn [-dxy] [-fn] [-pn] [-sn] [-] [file]...
```

netstat

```
netstat [-a] [-A] [-n] [-f address-family] [kernel]
[memory-interface]
netstat [-h] [-i] [-m] [-n] [-r] [-s] [-f address-family] [kernel]
[memory-interface]
netstat [-I interface] interval [kernel] [memory-interface]
netstat -I interface [-n] [kernel] [memory-interface]
```

newaliases

```
newaliases
```

newconfig

```
newconfig [-k] [module]... [nomodule]... [nonet] [-v]
```

newform

newform [-an] [-bn] [-cchar] [-en] [-f] [-itabspec] [-ln]
 [-otabspec] [-pn] [-s] [file]...

newfs

newfs [-b *block-size*] [-c *cylinders-per-group*] [-f *fragment-size*]
 [-i *bytes-per-inode*] [-m *free-space*] [-r *revolutions-per-minute*]
 [-s *size*] [-t *tracks-per-cylinder*] [-v] *device-file type*

newgrp

newgrp [-] [group]

news

news [-a] [-n] [-s] [items]

newunix

newunix [[no]module]...

nfsd

nfsd [*nserver*]...
 biod [*nserver*]...

nfsstat

nfsstat [-c] [-n] [-r] [-s] [-z]

nice

nice [-increment] *command* [*arguments*]

n1

n1 [-btype] [-ddelim] [-ftype] [-htype] [-iincr] [-lnum] [-nformat]
 [-p] [-ssep] [-vstart#] [-wwidth] *file*

nm

nm [-d] [-e] [-f] [-h] [-n] [-o] [-T] [-u] [-v] [-V] [-x] *file*...

nohup

nohup *command-line* &

nroff

nroff [-e] [-h] [-i] [-mname] [-nstart-no] [-opage-range] [-q]
 [-rletter[*value*]] [-s[*pages-per-pause*]] [-Ttty-type] [-u[*boldening-amt*]]
 [-z] [file]...

nslookup

```
nslookup
nslookup -server
nslookup host-to-find [server]
```

nterm**nulladm**

See `acctsh`.

number

```
number
```

od

```
od [-b] [-c] [-d] [-o] [-s] [-x] [file] [[+]offset [. ] [b]]
```

otroff

```
otroff [-cname] [-b] [-f] [-kname] [-mname] [-ppoint-size] [-t]
[-w] [file]...
```

pac

```
pac [-c] [-m] [-pprice] [-Pprinter] [-r] [-s] [name]...
```

pack

```
pack [-] [-f] file...
pcat file...
unpack file...
```

page

See `more`.

pagesize

```
pagesize
```

passwd

```
passwd [name]
```

paste

```
paste file1 file2 ...
paste -dlist file1 file2 ...
paste -s [-dlist] file1 file2 ...
```

pax

pax [-cimopuvy] [-f *archive*] [-s *replstr*] [-t *device*] [*pattern*]...
 pax -r [-cimnopuvy] [-f *archive*] [-s *replstr*] [-t *device*]
 [*pattern*]...
 pax -w [-adimuvy] [-b *blocking*] [-f *archive*] [-s *replstr*]
 [-t *device*] [-x *format*] [*path*]...
 pax -rw [-ilmopuvy] [-s *replstr*] [*path*]... *directory*

pcat

See pack.

pdp11

See machid.

pg

pg [-number] [+linenumber] [+/*pattern*] [-c] [-e] [-f] [-n]
 [-p *string*] [-s] [*file*]...

pic

pic [-T*tty-type*] [-] [*file*]...

ping

ping [-d] [-r] [-v] *host* [*packet-size*] [*npackets*]

pname

pname [-a] [-c *controller*] [-d *disk*] [-s *slice*] [-t *type*] *name*
 pname [-p]
 pname -a [*v*]
 pname -u *device-file*...

portmap

portmap

powerdown

powerdown

powerfail

See brc.

pr

pr [+pageno] [-columns] [-a] [-d] [-eck] [-f] [-h *head*] [-ick] [-lk]
 [-m] [-nck] [-ok] [-p] [-r] [-sc] [-t] [-wk] [*file*]...

prctmp

See acctsh.

prdaily

See acctsh.

printenv

printenv [*argument*]

prof

prof [-a] [-c] [-g] [-h] [-m *mdata*] [-n] [-o] [-s] [-t] [-x] [-z]
[*objfile*]

prof

```
#define MARK
#include <prof.h>
void MARK (name)
```

prs

prs [-a] [-c[*date-time*]] [-d[*dataspec*]] [-e] [-l] [-r[*SID*]] *file...*

prtacct

See acctsh.

ps

ps [-a] [-c*corefile*] [-d] [-e] [-f] [-g*grplist*] [-l] [-n*namelist*]
[-p*proclist*] [-s*swapdev*] [-t*termlist*] [-u*uidlist*]

psbanner

See transcript.

pscomm

See transcript.

psdit

psdit [-F *fontdir*] [-o *list*] [-p *prologue*] [*file*]

psinterface

See transcript.

psroff

psroff [-t] [[-a] [-i] [-mname] [-nN] [-olist] [-q] [-raN] [-sM]
 [-Tdest]] [[-ddest] [-C class] [-J name] [-h] [-nx] [-P printer] [-r]
 [-s] [-m] [-w]] [file]...

psrv

See transcript.

pstat

pstat [-a] [-b] [-f] [-i] [-m] [-nname] [-p] [-rrate] [-t]
 [-uaddress] [-v] [file]

pstext

See transcript.

ptx

ptx [-b break] [-f] [-g gap] [-i ignore] [-r] [-t] [-w n]
 [input [output]]
 ptx [-b break] [-f] [-g gap] [-o only] [-r] [-t] [-w n]
 [input [output]]

pwck

pwck [file]
 grpck [file]

pwd

pwd

query

query [-t[seconds]] [-r[response]] [-m]

quiz

quiz [-ifile] [-t] [category1 category2]

rain

rain

rc

See brc.

rcp

rcp file1 file2
 rcp [-r] file... directory

rcs

rsc [-alogins] [-Aoldfile] [-cstring] [-e[logins]] [-i] [-l[rev]] [-L] [-nname[:rev]] [-Nname[:rev]] [-orange] [-q] [-sstate[:rev]] [-t[txtfile]] [-u[rev]] [-U] files

rscdiff

rscdiff [-b] [-c] [-e] [-f] [-h] [-i] [-n] [-t] [-w] [-rrev1] [-rrev2] file ...

rscintro**rscmerge**

rscmerge -rrev1 [-rrev2] [-p] file

rcvhex

rcvhex [-p port] [-c command] file

rdist

rdist [-b] [-dvar=value] [-fdistfile] [-h] [-i] [-mhost] [-n] [-q] [-R] [-v] [-w] [-y] [name]...
rdist [-b] -c name... [-h] [-i] [-n] [-q] [-R] [-v] [-w] [-y] [login@] host [:dest]

rdump

See dump.bsd.

read_disk

read_disk

reboot

reboot [-h] [-l] [-n] [-q]

red

See ed.

refer

refer [-a[n]] [-b] [-B[l.m]] [-c keys] [-e] [-fn] [-kx] [-l[m,n]] [-n] [-p bib] [-P] [-s keys] [-S] [file]...

regcmp

regcmp [-] file...

regex

```
#define INIT declarations
#define GETC() getc-code
#define PEEKC() peekc-code
#define UNGETC (c) ungetc-code
#define RETURN (pointer) return-code
#define ERROR (val) errors-code
#include <regex.h>
char *compile(instring, expbuf, endbuf, eof)
char *instring, *expbuf, *endbuf;
int eof ;
int step(string, exbuf)
char *string, *exbuf;
extern char *loc1, *loc2, *locs;
extern int circf, sed, nbra;
```

reject

```
reject [-r reason] [destination]...
```

remlogin

```
remlogin -h host-name terminal-type
remlogin -h host-name -p
remlogin -r host-name
```

remsh

```
remsh rhost [-l username] [-n] [command]
```

remshd

```
in.remshd host.port
```

reset

```
See tset.
```


restore

restore i [b] [f] [F] [h] [m] [s] [v] [y] [*bfFs-arg*]...

restore r [b] [f] [F] [h] [m] [s] [v] [y] [*bfFs-arg*]...

restore R [b] [f] [F] [h] [m] [s] [v] [y] [*bfFs-arg*]...

restore t [b] [f] [F] [h] [m] [s] [v] [y] [*bfFs-arg*]... [*archived-file*]...

restore x [b] [f] [F] [h] [m] [s] [v] [y] [*bfFs-arg*]... [*archived-file*]...

rrestore i [b] [f] [F] [h] [m] [o] [s] [-Ttype] [v] [y] [*bfFs-arg*]...

rrestore r [b] [f] [F] [h] [m] [o] [s] [-Ttype] [v] [y] [*bfFs-arg*]...

rrestore R [b] [f] [F] [h] [m] [o] [s] [-Ttype] [v] [y] [*bfFs-arg*]...

rrestore t [b] [f] [F] [h] [m] [o] [s] [-Ttype] [v] [y] [*bfFs-arg*]...
[*archived-file*]...

rrestore x [b] [f] [F] [h] [m] [o] [s] [-Ttype] [v] [y] [*bfFs-arg*]...
[*archived-file*]...

rev

rev [*file*]...

revnetgroup

revnetgroup [-h] [-u]

rexecd

in.rxecd *host* .*port*

rez

rez [-a] [-align *word-type*] [-c *reator*] [-d *macro-assignment*]...
[-i *include-dir*]... [-o *output-file*] [-ov] [-p] [-rd] [-ro]
[-s *res-include-dir*]... [-t *type*] [-u *macro*]
[*resource-description-file*]...

rlog

rlog [-ddates] [-h] [-l[*lockers*]] [-L] [-rrevisions] [-R] [-sstates]
[-t] [-w[*logins*]] *file*...

rlogin

rlogin *rhost* [-8] [-ec] [-l *username*]

rlogind

in.rlogind *host.port*

rm

rm [-f] [-i] [-r] *file...*

rmdir *dir...*

rmail

rmail [-D*domain-name*] [-T] *login-name ...*

rmdel

rmdel -r *SID* [*file*]...

rmdir

See rm.

robots

robots

roffbib

roffbib [-e] [-h] [-m *name*] [-n*start-no*] [-o*page-range*]

-r*letter*[*integer*] [-s*N*] [-T*TTY-type*] [-x] [*file*]...

route

route [-f] [-n] *command* [*net* | *host*] *destination gateway* [*metric*]

routed

in.routed [-d] [-g] [-q] [-s] [-t] [*logfile*]

rpc.lockd

rpc.lockd [-g *grace-period*] [-t *timeout*]

rpc.statd

rpc.statd

rpc.yppasswdd

See yppasswdd.

rpcgen

```
rpcgen input-file
rpcgen -c [-o output-file] [input-file]
rpcgen -h [-o output-file] [input-file]
rpcgen -l [-o output-file] [input-file]
rpcgen -m [-o output-file] [input-file]
rpcgen -s transport [-o output-file] [input-file]
```

rpcinfo

```
rpcinfo -p [host]
rpcinfo -u host program-number version-number
rpcinfo -t host program-number version-number
```

rrestore

See `restore`.

rsh

See `sh`.

rstatd

```
rpc.rstatd
```

runacct

```
runacct [mmdd] [state]
```

rup

```
rup [-h] [-l] [-t] [host]...
```

ruptime

```
ruptime [-a] [-l] [-t] [-u]
```

rusers

```
rusers [-a] [-h] [-i] [-l] [-u] [host]...
```

rusersd

```
rpc.rusersd
```

rwall

```
rwall hosts
rwall -n netgroup...
rwall -h host -n netgroup
```

rwalld

rpc.rwalld

rwho

rwho [-a]

rwhod

in.rwhod

sa1

See sadc.

sa2

See sadc.

sactsact [-] *file*...**sadc**sadc [*t n*] [*file*]sa1 [*t n*]sa2 [-a] [-A] [-b] [-c] [-etime] [-i sec] [-m] [-q] [-stime] [-u] [-v]
[-w] [-y]**sag**sag [-e *time*] [-f *file*] [-i *sec*] [-s *time*] [-T *term*] [-x *spec*]
[-y *spec*]**sar**sar [-a] [-A] [-b] [-c] [-m] [-q] [-u] [-v] [-w] [-y] [-ofile] *t [n]*sar [-a] [-A] [-b] [-c] [-etime] [-f *file*] [-i sec] [-m] [-q] [-stime]
[-u] [-v] [-w] [-y]**sccs**sccs *command* [*flags*] [*args*] [-dpath] [-ppath] [-r]**sccsdiff**sccsdiff -rdelta1 -rdelta2 [-p] [-sn] *file*...**sccstorcs**sccstorcs [-t] [-v] *sccsfile* ...

script

script [-a] *file*

sdb

sdb [-w] [-W] [*objfile* [*corfile* [*directory*]]]

sdiff

sdiff [-l] [-o *output*] [-s] [-w *cols*] *file1 file2*

sed

sed [-n] -e *command-line-script* [*file*] ...

sed [-n] -f *scriptfile* [*file*] ...

sendmail

sendmail -bd -bi -bm -bp -bs -bt -bv -bz -C*configuration-file*
 -d*debug-level* -F*full-name* -fname -hhop-count -n
 -o*configuration-option value* -q[*interval*] -rname -t -v [*address*]...

setfile

setfile [-a*attribute-string*] [-c*creator*]

[-l*horizontal-pixels, vertical-pixels*] [-t*type*] [*data-file*]...

setport

setport -o [-s *baud-rate*] *tty*...

setport -r [-s *baud-rate*] *tty*...

settimezone

settimezone

sh

sh [-c *string*] [-i] [-r] [-s] [-a] [-e] [-f] [-h] [-k] [-n] [-t] [-u]

[-v] [-x] [*args*]...

rsh [-c *string*] [-i] [-r] [-s] [-a] [-e] [-f] [-h] [-k] [-n] [-t] [-u]

[-v] [-x] [*args*]...

shl

shl

showmount

showmount [-a] [-d] [-e] [*host*]

shutacct

See acctsh.

shutdown

shutdown [-ginterval] [-h] [-iinitstate] [-k] [-n] [-r] [-y]
[timeout [warning-message]]

size

size [-d] [-o] [-V] [-x] file...

slattach

slattach [+c] [-c] [+e] [-e] [+i] [-i] tty local-name remote-name
[baud-rate]

slattconf

slattconf [+c] [-c] [+e] [-e] [+i] [-i] tty baud-rate client-address
cslip-server-address [ifconfig-argument]...

sleep

sleep time

slip

slip

sno

sno [file]...

soelim

soelim [file]...

sort

sort [-b] [-c] [-d] [-f] [-i] [-m] [-M] [-n] [-o output] [-r] [-tx]
[-u] [-y [kmem]] [-zrecsz] [+pos1 [-pos2]] [file...]

sortbib

sortbib [-skeys] database...

spell

spell [-v] [-b] [-x] [-l] [+local-file] [file]...
hashmake
spellin n
hashcheck spelling-list

spellin

See spell.

spline

spline [-a] [-k] [-n] [-p] [-xlower *upper*]

split

split [-] [-n] [*file* [*output-file*]...]

spray

spray *host* [-c *count*] [-l *length*]

sprayd

rpc.sprayd

ssp

ssp [-] [*name*]...

startmac

startmac [-f *findername*] [-m *memsize*] [-o *name*[=*value*]]
 [-P *patchfile*] [-s *sysfolder*] [-S *systemfile*]
 startmac24 [-f *findername*] [-m *memsize*] [-o *name*[=*value*]]
 [-P *patchfile*] [-s *sysfolder*] [-S *systemfile*] [-u *user*] *startup-app*

startmac24

See startmac.

StartMonitor

StartMonitor

startmsg

startmsg -
 startmsg [-d *pcntdone*] [-m *msgselector*] [-n *nextphase*]
 [-p *numphases*] [-q] [*substr1*...*substr4*]

startup

startup

startup

See acctsh.

StartupShell**stat**

```
#include <sys/types.h>
#include <sys/stat.h>
```

statd

See `rpc.statd`.

stdhosts

`stdhosts file`

strings

`strings [-] [-o] [-number] file...`

strip

`strip [-l] [-r] [-s] [-V] [-x] file...`

stty

`stty [-a] [-g] [-n file] [options]`

style

`style [-a] [-e] [-l num] [-ml] [-mm] [-p] [-P] [-r num] file...`

su

`su [-] [name[arg ...]]`

subj

`subj file...`

sum

`sum [-r] file...`

sumdir

`sumdir [directories]`

swap

`swap -a [swapdev [swaplow [swaplen]]]`

`swap -d swapdev [swaplow]`

`swap -l`

sync

`sync`

sysinitrc

See `brc`.

sysline

```
sysline [+seconds] [-b] [-c] [-d] [-D] [-e] [-h] [-H remote] [-i]
[-j] [-l] [-m] [-p] [-q] [-r] [-s]
```

syslogd

```
syslogd [-d] [-fconfigfile] [-mmarkinterval]
```

systemfolder

```
systemfolder [-f] [-u user]
systemfolder24 [-f] [-u user]
```

systemfolder24

See systemfolder.

tabs

```
tabs [tabspec] [+m[n]] [-Ttype]
```

tail

```
tail [±[number][[b][f]]] [file]
tail [±[number][[c][f]]] [file]
tail [±[number][[l][f]]] [file]
```

talk

```
talk user [ttyname]
```

talkd

```
in.talkd
```

tar

```
tar [-]c[0...7[density]][ilvBdfs] [bBdfs-arg]... file...
tar [-]r[0...7[density]][ilvBdfs] [bBdfs-arg]... file...
tar [-]t[0...7][ivw][f archive] [file-in-archive]...
tar [-]u[0...7[density]][ilvBdfs] [bBdfs-arg]... file...
tar [-]x[0...7][timovw[f archive] [file-in-archive]...
```

tbl

```
tbl [-TX] [file]...
```

tc

```
tc [-a n] [-e] [-o list] [-t] [file]...
```

tcb

```
command-line | tcb >/dev/rmt/tcx
```

tcp

```
#include <sys/socket.h>
#include <netinet/in.h>
s = socket(AF_INET, SOCK_STREAM, 0);
```

tee

```
tee [-i] [-a] [file]...
```

telinit

See `init`.

telnet

```
telnet host [port]
telnet
```

telnetd

```
in.telnetd
```

term**test**

```
test [expr]
```

TextEditor

```
TextEditor [file]...
```

tftp

```
tftp [host]
```

tftpd

```
in.tftpd [-d] [-s] [home-directory]
```

tic

```
tic [-v[n]] file...
```

time

```
time command
```

timex

```
timex [-o] [-p[fhkmrt]] [-s] command
```

tip

tip [-v] [-speed] *system-name*
 tip [-v] [-speed] *phone-number*

touch

touch [-a] [-c] [-m] [*mmddhhmm*[*yy*]] *file...*

tp

tp d[[0...7] [i] [m] [v] [w]] [*file-in-archive*]...
 tp r[[0...7] [c] [i] [m] [v] [w]] [*file-in-archive*]...
 tp t[[0...7] [i] [m] [v] [w]] [*file-in-archive*]...
 tp u[[0...7] [c] [i] [m] [v] [w]] [*file-in-archive*]...
 tp x[[0...7] [f] [i] [m] [v] [w]] [*file-in-archive*]...

tplot

tplot [-T*terminal* [-e *raster-file*]]

tput

tput [-T*type*] *capname*

tr

tr [-c] [-d] [-s] [*string1* [*string2*]]

transcript

psbanner
 pscomm
 psinterface
 psrv
 pstext

trek

trek [[-a] *file*]

troff

troff [-] [-a] [-i] [-m*name*] [-n*M*] [-olist] [-q] [-ra*N*] [-s*M*]
 [-T*dest*] [*file...*]

troff**trpt**

trpt [-a] [-j] [-p*hex-address*] [-s] [-t] [*system*[*core*]]

true

```
true
false
```

tset

```
tset [-] [-a type] [-A] [-d type] [-ec] [-Ec] [-kc] [-l] [-m port]
[-p type] [-Q] [-r] [-s] [-S]
reset
```

tsort

```
tsort [file]
```

ttt

```
ttt
cubic
```

tty

```
tty [-l] [-s]
```

tty_add

```
tty_add [-r] [-glabel] device-file-name...
tty_kill
```

tty_kill

See `tty_add`.

tunefs

```
tunefs [-a maxcontig] [-d rotdelay] [-e maxbpg] [-m free-space]
[-o optimization] [-p] disk-device-file
```

turnacct

See `acctsh`.

twinkle

```
twinkle [-] [+] [s file] [density1 [density2]]
```

types

```
#include <sys/types.h>
```

tzdump

```
tzdump [-c cutoffyear] [-v] [zone]...
```

tzic

tzic [-d *directory*] [-l *localtime-link*] [-L *leap-file*]
[-p *posixrules-link*] [-s] [-v] [*source-file*]...

u3b

See machid.

u3b15

See machid.

u3b2

See machid.

u3b5

See machid.

ucbdiff

ucbdiff [-b] [-c] [-e] [-f] [-h] [-i] [-l] [-n] [-r] [-s] [-S *file*]
[-t] [-w] *dir1 dir2*
ucbdiff [-b] [-c] [-e] [-f] [-h] [-i] [-n] [-t] [-w] *file1 file2*
ucbdiff [-b] [-D*string*] [-i] [-w] *file1 file2*

ucbdiff3

ucbdiff3 [-e] [-E] [-x[-3]] [-X[-3]] *ver1 ver2 ver3*

udp

```
#include <sys/socket.h>
#include <netinet/in.h>
s=socket(AF_INET, SOCK_DGRAM, 0);
```

ul

ul [-t *terminal*] [*file*]...

umount

See mount.

uname

uname [-a] [-m] [-n] [-r] [-s] [-v]

uncompact

See compact.

uncompress

See compress.

uncompressdir

See compress.

unexpand

See expand.

unget

unget [-n] [-rSID] [-s] *file...*

uniq

uniq [-c] [-d] [+num] [-num] [-u] [*infile* [*outfile*]]

units

units

unpack

See pack.

updater

updater [d] [r] [u] *local remote...*

updater [p] [r] [u] *local remote...*

updater [t] [r] [u] *local remote...*

uptime

uptime

users

users [*file*]

uucheck

uucheck [-v] [-xdebug-level]

uucico

uucico [-cdevice-type] [-dspool-directory] [-f] [-iinterface]

[-rmode] [-ssystem] [-ulogin-name] [-xdebug-level]

uucleanup

uucleanup [-Cdays] [-Ddays] [-mstring] [-odays] [-ssystem]

[-Wdays] [-Xdays] [-xdebug-level]

uucp

uucp [-c] [-C] [-d] [-f] [-ggrade] [-j] [-m] [-nlogin-name] [-r]
[-sfile] [-xdebug-level] *source-file destination-file*

uucpd

/etc/uucpd

uudecode

See uuencode.

uudemon.admin

uudemon.admin

uudemon.cleanup

uudemon.cleanup

uudemon.hour

uudemon.hour

uudemon.poll

uudemon.poll

uuencode

uuencode [*source-file*] *decoded-name*
uudecode [*encoded-file*]

uuglist

uuglist [-l] [-u] [-xdebug-level]

uulog

uulog [-cq x] [-l[*hours*]] [-lines] [-fsystem] [*system*]...
uulog [-cq x] [-l[*hours*]] [-lines] [-ssystem] [*system*]...

uuname

uuname [-c] [-l]

uupick

See uuto.

uusched

uusched [-udebug-level] [-xdebug-level]

uuseed

uuseed [-m *file-permission*] -r *sourcefile system1 !...remotefile*

uustat

uustat [-a] [-S*job-status*] [[-j] [-ssystem]] [-u*login-name*]
[-x*debug-level*]

uustat -k*job-id* [-n] [-x*debug-level*]

uustat -m [-x*debug-level*]

uustat -p [-x*debug-level*]

uustat -q [-x*debug-level*]

uustat -r*job-id* [-n] [-x*debug-level*]

uustat -t*system* [-d*minutes*] [-c] [-x*debug-level*]

uuto

uuto [-m] [-p] *file... destination*

uupick [-ssystem]

Uutry

Uutry [-c*device-type*] [-r] [-x*debug-level*] *system*

uux

uux [-] [-a*aname*] [-b] [-C] [-c] [-g*grade*] [-j] [-n] [-p] [-r]
[-s*file*] [-x*debug-level*] [-z] *command-string*

uuxqt

uuxqt [-ssystem] [-x*debug-level*]

val

val-

val [-m*name*] [-r*SID*] [-s] [-y*type*] *file...*

values

#include <values.h>

vax

See machid.

vc

vc [-a] [-c*char*] [-s] [-t] [*keyword=value*]...

vedit

See vi.

version

version *file*...

vi

vi [+*command*] [-l] [-r *file*] [-R] [-t *tag*] [-wn] [-x] *name*...

view [+*command*] [-l] [-r *file*] [-R] [-t *tag*] [-wn] [-x] *name*...

vedit [+*command*] [-l] [-r *file*] [-R] [-t *tag*] [-wn] [-x] *name*...

view

See vi.

vipw

vipw

volcopy

volcopy [-a] [-b*density*] [-buf] [-feet*size*] [-reel*num*] [-s]

fsname special1 volname1 special2 volname2

labelit special [fsname volume [-n]]

w

w [-h] [-l] [-s] [-u] [*user*]

wall

wall

wc

wc [-*chunk-size*] [*file*]...

what

what [-s] *file*...

whatis

whatis *command*...

whereis

whereis [-b] [-B *dir* [-f]] [-m] [-M *dir* [-f]] [-s] [-S *dir* [-f]]

[-u] *file*...

which

which [*name*]...

who

who [-a] [-b] [-d] [-H] [-l] [-p] [-s] [-t] [-T] [-u] [*file*]
who -r [-d] [-l] [-p] [-u] [*file*]
who -q [*file*]
who am i
who am I

whoami

whoami

whodo

whodo

worm

worm [*size*]

worms

worms [-field] [-length *n*] [-number *n*] [-trail]

write

write *user* [*line*]

wtmpfix

See fwtmp.

wump

wump

xargs

xargs [-eofstr] [-ireplstr] [-l*number*] [-n*number*] [-p] [-ssize]
[-t] [-x] [*command* [*cmd-args*]]

xstr

xstr [-] [-c] [*file*]

yacc

yacc [-d] [-l] [-t] [-v] *grammar*

yes

yes [*expletive*]

ypbind

See ypserv.

ypcat

```
ypcat [-d domain-name] [-k] [-t] map-or-nick-name  
ypcat -x
```

ypinit

```
ypinit -m  
ypinit -s server-name
```

ypmake

```
cd /etc/yp; make [set-name] [variable=value...]
```

ypmatch

```
ypmatch [-d domain] [-k] [-t] key ... nickname-or-map-name  
ypmatch -x
```

yppasswd

```
yppasswd [login-name]
```

yppasswdd

```
rpc.yppasswdd file [-m make-arg...]
```

yppoll

```
yppoll [-h host] [-d domain] mapname
```

yppush

```
yppush [-d domain-name] [-v] mapname
```

ypserv

```
ypserv  
ypbind [-s] [-secure] [-v] [-ypset] [-ypsetme]
```

ypset

```
ypset [-V1] [-d domain-name] [-h host-name] server  
ypset [-V2] [-d domain-name] [-h host-name] server
```

ypwhich

```
ypwhich [-d domain-name] [-V1] [host-name]  
ypwhich [-d domain-name] [-V2] [host-name]  
ypwhich [-d domain-name] [[-t] -m [map-or-nickname]]  
ypwhich -x
```

ypxfr

`ypxfr [-c] [-d domain-name] [-f] [-h host-name] map-name`
`ypxfr -d domain-name [-C tid prog ipadd port] map-name`

zcat

See `compress`.

zcmp

See `compress`.

zdiff

See `compress`.

zmore

See `compress`.

A/UX Reference Summary and Index was written, edited, and composed on a desktop publishing system using Apple Macintosh computers, and `troff` running on A/UX. Page proofs were created on Apple LaserWriter printers. Final pages were output directly to 70-mm film on an Electrocomp 2000 Electron Beam Recorder. PostScript, the page-description language for the LaserWriter, was developed by Adobe Systems Incorporated.

Text type and display type are Times, Garamond, and Helvetica. Bullets are ITC Zapf Dingbats®. Some elements, such as program listings, are set in Apple Courier, a fixed-width font.

Writers: Mike Elola and Kathy Wallace

Developmental Editor: Paul Dreyfus

Art Director: Tamara Whiteside

Production Editor: Jeannette Allen

Production Supervisor: Robin Kerns

Special thanks to Michael Hinkson and George Towner

Index

absolute value

abs(3C) — return integer absolute value

abs(3F) — Fortran absolute value

floor(3M) — floor, ceiling, remainder, absolute value functions

access lists, groups

getgroups(2) — gets group access list

setgroups(2) — sets group access list

initgroups(3) — initializes group access list

access, time of last

utime(2) — set file access and modification times

accounting, connect-time

acctcon(1M) — invoke connect-time accounting

fwtmp(1M) — manipulate connect accounting records

accounting, of processes

ipcs(1) — reports interprocess communication facilities status

lav(1) — displays load average statistics

sag(1G) — generates a system activity graph

sar(1) — reports system activity

sysline(1) — displays the system status on the status line of a terminal

acct(1M) — present an overview of accounting commands

acctcms(1M) — summarizes commands from per-process accounting records

acctcom(1M) — searches and formats process accounting files

acctcon(1M) — invoke connect-time accounting

acctmerg(1M) — merges or adds accounting files

acctprc(1M) — provide process accounting

acctsh(1M) — provide shell procedures for accounting

diskusg(1M) — generates disk accounting data by user ID

fwtmp(1M) — manipulate connect accounting records

pac(1M) — gathers printer/plotter accounting information

runacct(1M) — runs daily accounting

sadc(1M) — report system activity

acct(2) — enable or disable process accounting

times(2) — get process and child process times

acct(4) — per-process accounting file format

prof(5) — profile within a function

accounting, system

ipcs(1) — reports interprocess communication facilities status

lav(1) — displays load average statistics

sag(1G) — generates a system activity graph

sar(1) — reports system activity

sysline(1) — displays the system status on the status line of a terminal
 acct(1M) — present an overview of accounting commands
 acctcms(1M) — summarizes commands from per-process accounting records
 acctcom(1M) — searches and formats process accounting files
 acctcon(1M) — invoke connect-time accounting
 acctmerg(1M) — merges or adds accounting files
 acctprc(1M) — provide process accounting
 acctsh(1M) — provide shell procedures for accounting
 diskusg(1M) — generates disk accounting data by user ID
 fwtmp(1M) — manipulate connect accounting records
 pac(1M) — gathers printer/plotter accounting information
 runacct(1M) — runs daily accounting
 sadc(1M) — report system activity

address handling

arp(1M) — displays and modifies the address translation table
 stdhosts(1M) — converts Internet addresses to standard form
 end(3C) — last locations in program

advisory lock

flock(2) — applies or removes an advisory lock on an open file

alarm clock

alarm(2) — sets a process's alarm clock

alert dialog boxes

macquery(1M) — posts a Macintosh alert box to query the user

aliases, mail

newaliases(1M) — rebuilds the database for the mail aliases file
 aliases(4) — address and alias format used by sendmail

aliens

aliens(6) — plays the game of Space Invaders (A/UX version)

Apple ImageWriter

iw2(1) — prepares data to be printed on the Apple ImageWriter II printer

Apple Tape Backup SC40

tcb(1) — blocks data to 8K for direct input to /dev/rmt/tcx
 tc(7) — tape device driver

AppleTalk, administration

appleping(1M) — exercises the AppleTalk network by sending packets to a named host
 appletalk(1M) — enables you to configure and display AppleTalk network interfaces

AppleTalk Datagrams

ddp(3N) — provide an AppleTalk Datagram Delivery Protocol (DDP) interface
 udp(5P) — Internet User Datagram Protocol

AppleTalk, printing with

- `at_cho_prn(1)` — allows you to choose a default printer on the AppleTalk internet
- `atlookup(1)` — looks up network-visible entities (NVEs) registered on the AppleTalk network system
- `atprint(1)` — transfers data to a printer by using AppleTalk protocols
- `atstatus(1)` — displays status information from an AppleTalk device

AppleTalk Transaction Protocol

- `atp(3N)` — provide a AppleTalk Transaction Protocol (ATP) interface

arccosine

- `acos(3F)` — Fortran arccosine intrinsic function
- `trig(3M)` — provide trigonometric functions

archive files

- `ar(1)` — maintains a library of files in an archive
- `cpio(1)` — copies files to or from a `cpio` archive
- `lorder(1)` — finds the ordering relation for an object library
- `pax(1)` — copies files to or from an archive in an IEEE format
- `tar(1)` — copies files to or from a `tar` archive
- `ldahread(3X)` — reads the archive header of a member of an archive file
- `ar(4)` — common archive file format
- `cpio(4)` — format of `cpio` archive
- `tar(4)` — format of `tar` header

arcsine

- `asin(3F)` — Fortran arcsine intrinsic function
- `trig(3M)` — provide trigonometric functions

arctangent

- `atan2(3F)` — Fortran arctangent intrinsic function
- `atan(3F)` — Fortran arctangent intrinsic function
- `trig(3M)` — provide trigonometric functions

arguments

- `apply(1)` — passes its arguments in batches to a command that is run once per every batch
- `echo(1)` — echoes its arguments
- `expr(1)` — evaluates arguments as an expression
- `xargs(1)` — builds arguments based on the standard input, passing them in batches to the specified command which is executed enough times to deplete all the arguments
- `getarg(3F)` — return Fortran command-line argument
- `getopt(3C)` — get option letter from argument vector
- `iargc(3F)` — return command line arguments
- `varargs(3X)` — handle variable argument list
- `vprintf(3S)` — format and output data from a variable-length argument list

arithmetic

- `bc(1)` — processes an arbitrary-precision arithmetic language
- `dc(1)` — desk calculator
- `expr(1)` — evaluates arguments as an expression
- `factor(1)` — prints the prime factor of a given number
- `units(1)` — rescales quantities according to a the unit of measure specified

ASA character set

- `asa(1)` — interprets ASA carriage control characters

ASCII character set

- `strings(1)` — finds the printable strings in an object or other binary file
- `a64l(3C)` — convert between long integer and base-64 ASCII string
- `atof(3C)` — converts an ASCII string to floating-point number
- `ctime(3)` — convert date and time to ASCII
- `ethers(3N)` — provide Ethernet address mapping operations
- `ascii(5)` — map of ASCII character set

assembly language

- `as(1)` — assembles files by translating assembler mnemonics to object code
- `cc(1)` — invokes the C compiler
- `dbx(1)` — debugs and executes programs
- `dis(1)` — produces an assembly language listing for a specified file

assertions

- `assert(3X)` — verify program assertion

ATP

- `atp(3N)` — provide a AppleTalk Transaction Protocol (ATP) interface

Autologic APS-5 phototypesetter

- `daps(1)` — invokes the Autologic APS-5 phototypesetter `troff` post-processor

autorecovery

- `escher(1M)` — helps you with autorecovery administration
- `eu(1M)` — updates autorecovery files
- `eupdate(1M)` — updates important files for autorecovery purposes
- `cml(4)` — configuration master list file format

backgammon

- `back(6)` — plays the game of backgammon

backing up files

- `cp(1)` — copies files
- `cpio(1)` — copies files to or from a `cpio` archive
- `pax(1)` — copies files to or from an archive in an IEEE format
- `tar(1)` — copies files to or from a `tar` archive
- `bcopy(1M)` — copies blocks interactively
- `dcopy(1M)` — copies System V File System-style file systems for optimal access time

dump.bsd(1M) — create a `dump.bsd` archive by making copies of files from a given file system
escher(1M) — helps you with autorecovery administration
eu(1M) — updates autorecovery files
eupdate(1M) — updates important files for autorecovery purposes
finc(1M) — generates a fast incremental backup for System V file systems
frec(1M) — recovers files from a backup tape
restore(1M) — retrieve files from within a `dump.bsd` archive into an existing file system
volcopy(1M) — copy file systems with label checking
dump.bsd(4) — format of a file-system dump
tc(7) — tape device driver

bad blocks

badblk(1M) — sets or updates bad block information
altblk(4) — alternate block information for bad block handling

banner printing

banner7(1) — generates a large banner
banner(1) — generates a poster

base portion of pathnames

basename(1) — get part of a pathname

base-64 numbers

a64l(3C) — convert between long integer and base-64 ASCII string

batch processing

at(1) — run commands at a later time
crontab(1) — aids in the use of the `cron` process scheduling program
env(1) — sets the environment for command execution
nice(1) — executes a command at low priority
nohup(1) — runs a command so that it can continue to run even after your session has ended
remsh(1N) — invokes to a shell on a remote system
shl(1) — manages the layering of multiple shells
yes(1) — generates `y` entries in response to requests for input
chroot(1M) — changes the root directory for a command
cron(1M) — runs the clock daemon

baud rate

stty(1) — sets the modes for a terminal
getty(1M) — set the initial communication modes, such as speed and line discipline, for the purpose of logging users in to A/UX through serial lines
cfgetospeed(3P) — get or set the value of the output and input baud rate

Berkeley Software Distribution, support for

- `sigvec(2)` — optional BSD-compatible software signal facilities
- `curses5.0(3X)` — provides BSD-style screen functions with optimal cursor motion
- `set42sig(3)` — sets the Berkeley Software Distribution (BSD) 4.2 signal interface

Bessel functions

- `bessel(3M)` — Bessel functions

bibliographies

- `addbib(1)` — creates or extends a bibliographic database
- `indxbib(1)` — builds an inverted index for a bibliography
- `lookbib(1)` — finds references in a bibliography
- `roffbib(1)` — prints out all records in a bibliographic database
- `sortbib(1)` — sorts bibliographic database

big files

- `bdiff(1)` — compares the difference between two large files that are too big for `diff` to handle
- `bfs(1)` — edits big files

binary-coded decimal

- `bcd(6)` — simulates a punched card corresponding to a text argument

binding

- `ypserv(1M)` — provide Network Information Service (NIS) service
- `bind(2N)` — bind a name to a socket
- `HOSTNAME(4)` — host name and domain name database

blackjack

- `bj(6)` — plays the game of black jack

blank lines in text

- `ssp(1)` — produces single spaced output

block zero information for file systems

- `bzb(4)` — Block Zero Block file format

blocking data

- `dd(1)` — converts and copies a file
- `tcb(1)` — blocks data to 8K for direct input to `/dev/rmt/tcx`

Boolean functions

- `test(1)` — evaluates conditions
- `true(1)` — provides truth values
- `bool(3F)` — Fortran bitwise boolean functions

Bourne shell

- `sh(1)` — runs the Bourne shell

bridges

- `rtmp(3N)` — identify AppleTalk node and bridge addresses

BSD, support for

- `sigvec(2)` — optional BSD-compatible software signal facilities
- `curses5.0(3X)` — provides BSD-style screen functions with optimal cursor motion
- `set42sig(3)` — sets the Berkeley Software Distribution (BSD) 4.2 signal interface

buffering

- `col(1)` — filters text containing printer control sequences for use at a display device
- `setbuf(3S)` — assign buffering to a stream

byte order

- `byteorder(3N)` — convert values between host and network byte order

C programming language

- `cb(1)` — improves spacing and indentation of C source files
- `cc(1)` — invokes the C compiler
- `cflow(1)` — generates a C flowgraph
- `cpp(1)` — invokes the C language preprocessor
- `ctags(1)` — maintains a tags file for a C program
- `ctrace(1)` — debugs a C program
- `cxref(1)` — generates a C program cross-reference
- `ident(1)` — displays RCS keywords and their values
- `indent(1)` — indents and formats C program source
- `lint(1)` — invokes a C program checker
- `mkshlib(1)` — creates a shared library
- `mkstr(1)` — creates an error message file by massaging C source programs
- `xstr(1)` — reports strings from C programs to implement shared strings

calculate

- `bc(1)` — processes an arbitrary-precision arithmetic language
- `dc(1)` — desk calculator
- `expr(1)` — evaluates arguments as an expression
- `factor(1)` — prints the prime factor of a given number
- `units(1)` — rescales quantities according to a the unit of measure specified

calendar

- `cal(1)` — displays a calendar
- `calendar(1)` — provides a reminder service

ceiling numbers

- `floor(3M)` — floor, ceiling, remainder, absolute value functions

change bars

- `diffmk(1)` — marks the differences between two files

character codes

`charcv(3C)` — converts the character code to another encoding scheme

character count

`sumdir(1)` — sums and counts the characters within the files of the given directories

`wc(1)` — counts characters, words, and lines in a file

character frequency

`freq(1)` — reports character frequencies in a file

characters, general

`cut(1)` — cuts out selected fields of each line of a file

`freq(1)` — reports character frequencies in a file

`paste(1)` — merges lines of several files or subsequent lines of one file

`rev(1)` — reverses characters within each line of text

`tr(1)` — translates characters

`wc(1)` — counts characters, words, and lines in a file

`charcv(3C)` — converts the character code to another encoding scheme

`conv(3C)` — translate characters

`ctype(3C)` — classify characters

`getc(3S)` — get character or word from a stream

`putc(3S)` — put a character or word on a stream

`ungetc(3S)` — pushes a character back into input stream

`eqnchar(5)` — special character definitions for `eqn` and `neqn`

`greek(5)` — graphics for the extended TTY-37 type-box

checksums

`sum(1)` — calculates a checksum

CML

`escher(1M)` — helps you with autorecovery administration

`eupdate(1M)` — updates important files for autorecovery purposes

`cml(4)` — configuration master list file format

code sections

`size(1)` — displays section sizes of common object files

columns

`colrm(1)` — removes columns from a file

`cut(1)` — cuts out selected fields of each line of a file

`paste(1)` — merges lines of several files or subsequent lines of one file

command interpretation, audit trail for

`script(1)` — starts a shell that records terminal input and output

command interpretation, windows for

`CommandShell(1)` — manages command-interpretation windows and moderates access to the A/UX console window

`vt102(7)` — provides protocols for VT102 terminals

command line generation

- `apply(1)` — passes its arguments in batches to a command that is run once per every batch
- `cmdo(1)` — builds command lines interactively
- `xargs(1)` — builds arguments based on the standard input, passing them in batches to the specified command which is executed enough times to deplete all the arguments

command line interpreters

- `csh(1)` — runs the C shell, a command interpreter with C-like syntax
- `ksh(1)` — runs the Korn shell, an enhanced command interpreter that is backward-compatible with the Bourne shell (`sh`)
- `remsh(1N)` — invokes to a shell on a remote system
- `sh(1)` — runs the Bourne shell
- `shl(1)` — manages the layering of multiple shells
- `StartupShell(8)` — interprets command lines such as those used to boot A/UX and check file systems within the A/UX Startup application

command options, help

- `cmdo(1)` — builds command lines interactively

command-line arguments

- `apply(1)` — passes its arguments in batches to a command that is run once per every batch
- `echo(1)` — echoes its arguments
- `expr(1)` — evaluates arguments as an expression
- `xargs(1)` — builds arguments based on the standard input, passing them in batches to the specified command which is executed enough times to deplete all the arguments
- `getarg(3F)` — return Fortran command-line argument
- `getopt(3C)` — get option letter from argument vector
- `iargc(3F)` — return command line arguments
- `varargs(3X)` — handle variable argument list
- `vprintf(3S)` — format and output data from a variable-length argument list

commands

- `apropos(1)` — locates commands by keyword
- `env(1)` — sets the environment for command execution
- `uux(1C)` — runs a command on a remote system
- `whatis(1)` — reports a brief description for the manual page entry specified
- `which(1)` — reports the directory path to a file by interpreting PATH and alias settings
- `system(3F)` — issues a shell command from Fortran
- `system(3S)` — issues a shell command

commands, device-specific

`clear(1)` — clears the terminal screen
`eject(1)` — ejects a diskette from the drive
`iw2(1)` — prepares data to be printed on the Apple ImageWriter II printer
`mt(1)` — manipulates magnetic tape media
`stty(1)` — sets the modes for a terminal
`tabs(1)` — sets the tab stops on a terminal
`tcbl(1)` — blocks data to 8K for direct input to `/dev/rmt/tcx`
`keyset(1M)` — sets the keyboard for the console

communicating, host-to-host through TCP/IP

`ftp(1N)` — transfers files by using the DARPA Internet File Transfer Protocol (FTP)
`rdist(1)` — distributes remote files
`remsh(1N)` — invokes to a shell on a remote system
`telnet(1C)` — communicates with another host via the TELNET protocol
`tftp(1C)` — transfers files via the Trivial File Transfer Protocol (TFTP)
`rwall(1M)` — writes to all users over a network
`slip(1M)` — assigns a serial line to a network interface
`tftpd(1M)` — responds to requests to use the DARPA Trivial File Transfer Protocol

communicating, through serial ports

`ct(1C)` — runs `login` on a dial-up line
`cu(1C)` — establishes an interactive connection with another system
`kermit(1C)` — invokes the Kermit file-transfer program
`tip(1C)` — establishes a connection to a remote system
`updater(1)` — updates files between two machines
`uencode(1C)` — encode and decode a binary file

communicating, through the UUCP system

`uucp(1C)` — copies files from one system to another system
`uuglist(1C)` — displays the service grades that are available on your system
`uusend(1C)` — sends a file to a remote host
`uustat(1C)` — controls `uucp` jobs and provides status information
`uuto(1C)` — provide an easy interface to the `uucp` command, using the public directories
`uux(1C)` — runs a command on a remote system

communicating, using AppleTalk

`at_cho_prn(1)` — allows you to choose a default printer on the AppleTalk internet
`atlookup(1)` — looks up network-visible entities (NVEs) registered on the AppleTalk network system
`atprint(1)` — transfers data to a printer by using AppleTalk protocols
`atstatus(1)` — displays status information from an AppleTalk device

communicating, utilities for

`biff(1)` — enables and disables notification of mail by `comsat`
`from(1)` — displays the mail header lines in your mailbox
`mail(1)` — send mail to users or read mail
`mailx(1)` — enables you to send and receive messages electronically
`mesg(1)` — permits or denies the receipt of messages
`news(1)` — displays local news items
`talk(1N)` — talks to another user via the terminal
`write(1)` — writes to another user
`wall(1M)` — writes to all users

comparing files and directories

`bdiff(1)` — compares the difference between two large files that are too big for `diff` to handle
`cmp(1)` — compares two files
`comm(1)` — selects or rejects lines common to two sorted files
`diff3(1)` — compares three versions of a file
`diff(1)` — compares two files or directories for any differences
`dircmp(1)` — compares the contents of two directories
`merge(1)` — merges three files into one
`rcsdiff(1)` — compares RCS revisions
`sccsdiff(1)` — compares two versions of an SCCS file
`sdiff(1)` — reports side-by-side differences between two files in a side-by-side format
`sumdir(1)` — sums and counts the characters within the files of the given directories
`ucbdiff3(1)` — reports the differences between three files
`ucbdiff(1)` — reports differences between two files or directories
`uniq(1)` — reports repeated lines in a file

compatibility

`setcompat(2)` — set or get process compatibility mode
`sigvec(2)` — optional BSD-compatible software signal facilities
`curses5.0(3X)` — provides BSD-style screen functions with optimal cursor motion
`set42sig(3)` — sets the Berkeley Software Distribution (BSD) 4.2 signal interface
`setposix(3P)` — sets POSIX compatibility flags

compilers

`bs(1)` — compiles and interprets `bs` programs
`cc(1)` — invokes the C compiler
`f77(1)` — invokes the Fortran 77 compiler
`regcmp(1)` — compiles regular expressions with a file
`rez(1)` — compiles Macintosh resource files from source code
`rpcgen(1)` — generates C source code from a remote procedure call (RPC) source file

sno(1) — runs the SNOBOL interpreter
 yacc(1) — compiles compilers (yet another compiler-compiler)
 tic(1M) — compiles (translates) terminfo files
 tzic(1M) — compiles time-zone information files that are required to set
 the local time-zone
 regcmp(3X) — compile and execute a regular expression
 regexp(5) — regular expression compile and match routines

complex numbers

aimag(3F) — Fortran imaginary part of complex argument
 conjg(3F) — Fortran complex conjugate intrinsic function

compressing and expanding files

compact(1) — compress and uncompress files
 compress(1) — compress files and directories as well as expand them;
 support concatenation, browsing, and file-comparing operations upon
 compressed files
 crypt(1) — encodes and decodes passwords
 makekey(1) — generates an encryption key
 pack(1) — compress and expand files

concatenation

cat(1) — catenates and displays the contents of files
 paste(1) — merges lines of several files or subsequent lines of one file

conditional execution

test(1) — evaluates conditions
 true(1) — provides truth values

configuration

checkinstall(1) — checks the installation of boards
 tset(1) — set or reset the terminal to a sensible state
 adduser(1M) — adds a user account
 autoconfig(1M) — creates an up-to-date kernel
 badblk(1M) — sets or updates bad block information
 chgnod(1M) — changes the current A/UX system node name
 diskformat(1M) — formats a disk through a driver-dependent format
 operation
 dp(1M) — performs disk partitioning
 getty(1M) — set the initial communication modes, such as speed and
 line discipline, for the purpose of logging users in to A/UX through
 serial lines
 init(1M) — spawn general processes
 kconfig(1M) — tunes kernel parameters for work-load optimization
 line_sane(1M) — pushes streams line disciplines
 lpadmin(1M) — configures the lp spooling system
 module_dump(1M) — queries kernel files for configuration information
 newconfig(1M) — generates an up-to-date kernel
 newunix(1M) — manipulates the files that determine the configuration of

a new kernel

`pname(1M)` — associates named partitions with device files
`pstat(1M)` — prints system facts
`setport(1M)` — sets the characteristics of a serial port
`settimezone(1M)` — sets the local time zone
`slattconf(1M)` — attaches a serial line to a network interface and configures the network interface
`swap(1M)` — adds disk blocks to or deletes them from the swap area
`tic(1M)` — compiles (translates) `terminfo` files
`tty_add(1M)` — modify the `/etc/inittab` file in terms of enabling serial ports for use as login terminals
`tzdump(1M)` — displays the date and time for one or more time zones
`tzic(1M)` — compiles time-zone information files that are required to set the local time-zone
`uvar(2)` — returns system-specific configuration information
`gettydefs(4)` — speed and terminal settings used by `getty`
`inittab(4)` — script for the `init` process
`master(4)` — master kernel-configuration file format

Configuration Master List

`escher(1M)` — helps you with autorecovery administration
`eupdate(1M)` — updates important files for autorecovery purposes
`cml(4)` — configuration master list file format

connect-time accounting

`acctcon(1M)` — invoke connect-time accounting
`fwtmp(1M)` — manipulate connect accounting records

connections

`cu(1C)` — establishes an interactive connection with another system
`telnet(1C)` — communicates with another host via the TELNET protocol
`tip(1C)` — establishes a connection to a remote system
`ping(1M)` — exercises the TCP/IP network by sending Internet Control Message Protocol (ICMP) packets to a named host
`accept(2N)` — accept a connection on a socket
`listen(2N)` — listens for connections on a socket
`shutdown(2N)` — shut down part of a full-duplex connection
`dial(3C)` — establishes an out-going terminal line connection
`lo(5)` — software loopback network interface

console

`keyset(1M)` — sets the keyboard for the console
`ioctl.syscon(4)` — console terminal settings file
`console(7)` — provides access to the console keyboard and screen

constant-width text

`cw(1)` — prepare constant-width text for `otroff`

constants

`values(5)` — machine-dependent values

converters

`conv(1)` — swaps bytes in COFF files

`dd(1)` — converts and copies a file

`enscript(1)` — converts text files to format for printing

`fcvt(1)` — converts a file in one storage format to a different storage format

`hex(1)` — converts an object file to Motorola S-record format

`mactoiso(1)` — convert between Macintosh encoding and International Standards Organization (ISO) encoding

`units(1)` — rescales quantities according to a the unit of measure specified

`a64l(3C)` — convert between long integer and base-64 ASCII string

copying

`atprint(1)` — transfers data to a printer by using AppleTalk protocols

`cp(1)` — copies files

`cpio(1)` — copies files to or from a `cpio` archive

`csplit(1)` — splits files into sections

`dd(1)` — converts and copies a file

`fcvt(1)` — converts a file in one storage format to a different storage format

`ln(1)` — makes links

`pax(1)` — copies files to or from an archive in an IEEE format

`rcp(1C)` — copies files between two systems

`split(1)` — splits a file into a specified number of pieces

`tar(1)` — copies files to or from a `tar` archive

`tp(1)` — copies files to or from a `tp` archive

`uucp(1C)` — copies files from one system to another system

`uuto(1C)` — provide an easy interface to the `uucp` command, using the public directories

`bcopy(1M)` — copies blocks interactively

`dcopy(1M)` — copies System V File System-style file systems for optimal access time

`dump.bsd(1M)` — create a `dump.bsd` archive by making copies of files from a given file system

`restore(1M)` — retrieve files from within a `dump.bsd` archive into an existing file system

`volcopy(1M)` — copy file systems with label checking

`blt(3C)` — block transfer data

`cpio(4)` — format of `cpio` archive

`tar(4)` — format of `tar` header

core image

fsync(2) — synchronize a file's in-core state with that on disk

core(4) — format of core image file

cosine

cos(3F) — Fortran cosine intrinsic function

cosh(3F) — Fortran hyperbolic cosine intrinsic function

trig(3M) — provide trigonometric functions

counters

sumdir(1) — sums and counts the characters within the files of the given directories

wc(1) — counts characters, words, and lines in a file

craps

craps(6) — plays the game of craps

crashes

errdead(1M) — extracts error records from a crash dump

statd(1M) — provide crash and recovery monitoring for network locking services

creating new objects

mkdir(1) — creates a directory

mkshlib(1) — creates a shared library

mkstr(1) — creates an error message file by massaging C source programs

mkfs1b(1M) — constructs a file system with 512-byte blocks

mkfs(1M) — constructs a System V file system

mklost+found(1M) — makes a directory named `lost+found` to be used by `fsck`

mknod(1M) — builds a device file

mkslipuser(1M) — creates or updates the Compressed Serial Line/Internet Protocol (CSL/IP) database

newconfig(1M) — generates an up-to-date kernel

newfs(1M) — makes a Berkeley 4.2 (UFS) file system

newunix(1M) — manipulates the files that determine the configuration of a new kernel

ypmake(1M) — rebuilds the Network Information Service (NIS) maps

creat(2) — creates a new file or rewrites an existing one

fork(2) — creates a new process

mkdir(2) — makes a directory file

mknod(2) — makes a directory, or a special or ordinary file

umask(2) — set and get file creation mask

mkfifo(3P) — makes a FIFO special file

mktemp(3C) — makes a unique filename

tmpfile(3S) — creates a temporary file

tmpnam(3S) — create a name for a temporary file

cribbage

cribbage(6) — plays the game of cribbage

cross-references

cxref(1) — generates a C program cross-reference

lorder(1) — finds the ordering relation for an object library

macref(1) — produces a cross-reference listing of macro files

current directory

pwd(1) — prints the name of the working directory

chdir(2) — changes the working directory

getcwd(3C) — gets the pathname of the current working directory

getwd(3) — gets the current working directory pathname

current host

gethostid(2N) — get/set unique identifier of current host

gethostname(2N) — get/set name of current host

current user

whoami(1) — prints effective current user ID

daemons

automount(1M) — mounts Network File System (NFS) when needed

cron(1M) — runs the clock daemon

errdemon(1M) — calls the error-logging daemon

errstop(1M) — terminates the error-logging daemon

inetd(1M) — starts Internet servers when needed

init(1M) — spawn general processes

lockd(1M) — handle local and remote lock requests

lpd(1M) — supports the Berkeley print spooler .“ 4.2 line-printer daemon

nfsd(1M) — invoke the NFS daemons

routed(1M) — invokes the network routing daemon

nfssvc(2) — provides NFS daemons

DARPA Internet

ftp(1N) — transfers files by using the DARPA Internet File Transfer Protocol (FTP)

nslookup(1) — interactively queries name servers

rmail(1) — handles remote mail received via UUCP

tftp(1C) — transfers files via the Trivial File Transfer Protocol (TFTP)

ftpd(1M) — provide Internet File Transfer Protocol (FTP) service

inetd(1M) — starts Internet servers when needed

named(1M) — provides Internet domain name service

portmap(1M) — converts RPC program numbers into DARPA protocol port numbers

sendmail(1M) — sends mail

stdhosts(1M) — converts Internet addresses to standard form

telnetd(1M) — supports the DARPA standard TELNET protocol

tftpd(1M) — responds to requests to use the DARPA Trivial File Transfer Protocol

- inet(3N) — provide Internet address manipulation routines
- resolver(3N) — provide resolver routines
- networks(4N) — network name database
- protocols(4N) — protocol name database
- resolv.conf(4) — configuration file for resolver routines
- servers(4) — Internet server database
- services(4N) — service name database
- arp(5P) — Address Resolution Protocol
- icmp(5P) — Internet Control Message Protocol
- inet(5P) — Internet protocol family
- ip(5P) — Internet Protocol
- tcp(5P) — Internet Transmission Control Protocol
- udp(5P) — Internet User Datagram Protocol

DASI 300 terminal

- 300(1) — filter text containing printer control sequences for a DASI terminal

DASI 450 terminal

- 450(1) — filters text containing printer control sequences for the DASI terminal

data, blocking of

- dd(1) — converts and copies a file
- tcb(1) — blocks data to 8K for direct input to /dev/rmt/tcx

data, redirecting

- cat(1) — catenates and displays the contents of files
- csh(1) — runs the C shell, a command interpreter with C-like syntax
- ksh(1) — runs the Korn shell, an enhanced command interpreter that is backward-compatible with the Bourne shell (sh)
- sh(1) — runs the Bourne shell
- tee(1) — transcribes data

data streams

- line_sane(1M) — pushes streams line disciplines
- fclose(3S) — close or flush a stream
- ferror(3S) — stream status inquiries
- fopen(3S) — open a stream
- fread(3S) — produce binary input/output
- fseek(3S) — reposition a file pointer in a stream
- getc(3S) — get character or word from a stream
- gets(3S) — get a string from a stream
- line_push(3) — routine used to push streams line disciplines
- printf(3S) — format and output string and numeric data
- putc(3S) — put a character or word on a stream
- puts(3S) — put a string on a stream
- rcmd(3N) — routines for returning a stream to a remote command
- rexec(3N) — returns a stream to a remote command

- scanf(3S) — convert formatted input
- setbuf(3S) — assign buffering to a stream
- ungetc(3S) — pushes a character back into input stream
- streams(7) — provides an interface for character I/O

data types

- ftype(3F) — explicit Fortran type conversion
- xdr(3N) — provide library routines for external data representation
- types(5) — primitive system data types

Datagrams

- ddp(3N) — provide an AppleTalk Datagram Delivery Protocol (DDP) interface
- udp(5P) — Internet User Datagram Protocol

date and time

- cal(1) — displays a calendar
- calendar(1) — provides a reminder service
- date(1) — displays and sets the date
- leave(1) — reminds you when you have to leave
- cron(1M) — runs the clock daemon
- settimezone(1M) — sets the local time zone
- gettimeofday(2) — get/set date and time
- stime(2) — set time
- time(2) — get time
- ctime(3) — convert date and time to ASCII
- tzfile(4) — time-zone information
- nvr(7) — provides an interface to nonvolatile memory

debuggers

- adb(1) — debugs executable programs
- ctrace(1) — debugs a C program
- dbx(1) — debugs and executes programs
- sdb(1) — symbolic debugger
- fsdb(1M) — debugs the file system
- ping(1M) — exercises the TCP/IP network by sending Internet Control Message Protocol (ICMP) packets to a named host
- lo(5) — software loopback network interface

decompiler

- derez(1) — decompiles a resource file

default values

- at_cho_prn(1) — allows you to choose a default printer on the AppleTalk internet
- chsh(1) — changes the default login shell
- umask(2) — set and get file creation mask
- finstallrc(4) — finstall default configuration file
- shells(4) — shell pathnames file

defaults, shell and session type

CommandShell(1) — manages command-interpretation windows and moderates access to the A/UX console window
chsh(1) — changes the default login shell
Login(1M) — logs you in to A/UX by using a graphical user interface
shells(4) — shell pathnames file

delayed execution

at(1) — run commands at a later time
crontab(1) — aids in the use of the cron process scheduling program
sleep(1) — suspends the system for a specified interval of time
cron(1M) — runs the clock daemon
pause(2) — suspends a process until signal

deleting

cancel(1) — cancels print requests spooled through the lp command
colrm(1) — removes columns from a file
cut(1) — cuts out selected fields of each line of a file
deroff(1) — removes nroff/troff, tbl, and eqn constructs
ipcrm(1) — removes interprocess communications facilities
kill(1) — terminates a process
lprm(1) — removes jobs from the line printer spooling queue for a Berkeley file system (4.2)
rm(1) — remove files or directories
rmdel(1) — removes a delta from an SCCS file
dev_kill(1M) — removes device files from a directory
killall(1M) — kills all active processes
flock(2) — applies or removes an advisory lock on an open file
rmdir(2) — remove a directory file
unlink(2) — remove directory entry
umount(2) — remove a file system
insque(3N) — insert/remove element from a queue

delta files (SCCS)

cdc(1) — changes the delta commentary of an SCCS delta
comb(1) — combines SCCS deltas
delta(1) — makes a delta (change) to an SCCS file
rmdel(1) — removes a delta from an SCCS file
sact(1) — displays who has checked a Source Code Control System (SCCS) file out for editing

description files, troff fonts

makedev(1) — prepares troff description files
afm(4) — Adobe POSTSCRIPT font metrics file format
font(5) — description files for device-independent troff

descriptor tables

`getdtablesize(2N)` — gets descriptor table size

descriptors, general

`close(2)` — closes a file descriptor

`dup(2)` — duplicates a descriptor

`dup2(3N)` — duplicates a descriptor

desktop, Macintosh

`CommandShell(1)` — manages command-interpretation windows and moderates access to the A/UX console window

device description files

`printcap(4)` — printer-capability database

`termcap(4)` — terminal capability database

`terminfo(4)` — terminal capability database

device file management

`tty(1)` — obtains the device filename for the terminal or CommandShell window where it is invoked

`dev_kill(1M)` — removes device files from a directory

`devnm(1M)` — displays the current device name

`mknod(1M)` — builds a device file

`pname(1M)` — associates named partitions with device files

`tty(7)` — controls the terminal interface

device files, overview

`intro(7)` — introduces device drivers and interfaces

device-specific commands

`clear(1)` — clears the terminal screen

`eject(1)` — ejects a diskette from the drive

`iw2(1)` — prepares data to be printed on the Apple ImageWriter II printer

`mt(1)` — manipulates magnetic tape media

`stty(1)` — sets the modes for a terminal

`tabs(1)` — sets the tab stops on a terminal

`tcb(1)` — blocks data to 8K for direct input to `/dev/rmt/tcx`

`keyset(1M)` — sets the keyboard for the console

Diablo 1620 printer

`450(1)` — filters text containing printer control sequences for the DASI terminal

dialog boxes, constructing Macintosh alert

`macquery(1M)` — posts a Macintosh alert box to query the user

dialog boxes, Macintosh

`cmdo(1)` — builds command lines interactively

`Login(1M)` — logs you in to A/UX by using a graphical user interface

`macquery(1M)` — posts a Macintosh alert box to query the user

dialup communication

`cu(1C)` — establishes an interactive connection with another system
`kermit(1C)` — invokes the Kermit file-transfer program
`tip(1C)` — establishes a connection to a remote system
`uucp(1C)` — copies files from one system to another system
`uux(1C)` — runs a command on a remote system
`slip(1M)` — assigns a serial line to a network interface
`uucico(1M)` — transfers files as specified by `uucp` work files
`dial(3C)` — establishes an out-going terminal line connection
`dialup(4)` — modem escape sequence file
`phones(4)` — remote host telephone number database

differences

`bdiff(1)` — compares the difference between two large files that are too big for `diff` to handle
`cmp(1)` — compares two files
`diff3(1)` — compares three versions of a file
`diff(1)` — compares two files or directories for any differences
`diffmk(1)` — marks the differences between two files
`dircmp(1)` — compares the contents of two directories
`rcsdiff(1)` — compares RCS revisions
`sccsdiff(1)` — compares two versions of an SCCS file
`sdiff(1)` — reports side-by-side differences between two files in a side-by-side format
`ucbdiff3(1)` — reports the differences between three files
`ucbdiff(1)` — reports differences between two files or directories

directories

`dircmp(1)` — compares the contents of two directories
`ln(1)` — makes links
`ls(1)` — lists the contents of a directory
`mkdir(1)` — creates a directory
`mv(1)` — moves or renames files
`sumdir(1)` — sums and counts the characters within the files of the given directories
`cpset(1M)` — installs files in specified directories
`dev_kill(1M)` — removes device files from a directory
`getdirentries(2)` — gets directory entries
`link(2)` — provides a link to a file
`mkdir(2)` — makes a directory file
`mknod(2)` — makes a directory, or a special or ordinary file
`rmdir(2)` — remove a directory file
`unlink(2)` — remove directory entry
`directory(3)` — perform operations on directories
`ftw(3C)` — walks a file tree
`scandir(3)` — scans a directory

dir(4) — format of System V directories
directory, current
 pwd(1) — prints the name of the working directory
 chdir(2) — changes the working directory
 getcwd(3C) — gets the pathname of the current working directory
 getwd(3) — gets the current working directory pathname
directory string functions
 basename(1) — get part of a pathname
 realpath(3) — returns the real filename of a file
disassembler
 dis(1) — produces an assembly language listing for a specified file
disk accounting
 df(1) — reports the used and unused storage capacity for a file system
 du(1) — summarizes disk usage
 diskusg(1M) — generates disk accounting data by user ID
disk blocks
 df(1) — reports the used and unused storage capacity for a file system
 du(1) — summarizes disk usage
 badblk(1M) — sets or updates bad block information
 bcopy(1M) — copies blocks interactively
 altblk(4) — alternate block information for bad block handling
 bzb(4) — Block Zero Block file format
disk drives
 eject(1) — ejects a diskette from the drive
disk partitions
 dd(1) — converts and copies a file
 dp(1M) — performs disk partitioning
 pname(1M) — associates named partitions with device files
 getptabent(3) — get partition table file entry
 bzb(4) — Block Zero Block file format
 dpme(4) — format of disk partition map entries
 ptab(4) — partition table file
disks, floppy
 cpio(1) — copies files to or from a cpio archive
 eject(1) — ejects a diskette from the drive
 pax(1) — copies files to or from an archive in an IEEE format
 tar(1) — copies files to or from a tar archive
 diskformat(1M) — formats a disk through a driver-dependent format
 operation
 finstall(1M) — installs A/UX software from specially prepared floppy
 disks
 cpio(4) — format of cpio archive
 finstallrc(4) — finstall default configuration file
 tar(4) — format of tar header

fd(7) — provides an interface to 3.5-inch disk drives

disks, formatting

diskformat(1M) — formats a disk through a driver-dependent format operation

disks, general

df(1) — reports the used and unused storage capacity for a file system

du(1) — summarizes disk usage

eject(1) — ejects a diskette from the drive

diskformat(1M) — formats a disk through a driver-dependent format operation

fsck(1M) — checks file-system consistency and interactively repairs the file system

fsync(2) — synchronize a file's in-core state with that on disk

disktab(4) — disk description file format

fstab(4) — parameter file format

gd(7) — provides a generic interface to disk devices

display processing

300(1) — filter text containing printer control sequences for a DASI terminal

4014(1) — filters text containing printer control sequences a page at a time

450(1) — filters text containing printer control sequences for the DASI terminal

col(1) — filters text containing printer control sequences for use at a display device

colcrt(1) — filters nroff output for terminal previewing

greek(1) — filters text for vintage display devices

tc(1) — interprets troff output for use at a vintage display device

tplot(1G) — interprets plotter instructions for use at a vintage display device

ul(1) — filters special underlining sequences imbedded in text for use at a display device

dividing files

csplit(1) — splits files into sections

split(1) — splits a file into a specified number of pieces

documentation, online

apropos(1) — locates commands by keyword

man(1) — displays the named manual page entries

whatis(1) — reports a brief description for the manual page entry specified

whereis(1) — reports the locations of the source, binary, and online help files for a specified command

man(5) — macros for formatting entries in this manual

domains

`domainname(1)` — sets or displays the name of the Network Information Service (NIS) domain
`named(1M)` — provides Internet domain name service
`resolver(3N)` — provide resolver routines
`HOSTNAME(4)` — host name and domain name database
`resolv.conf(4)` — configuration file for resolver routines

double-precision numbers

`aint(3F)` — Fortran integer part intrinsic function
`dprod(3F)` — Fortran double precision product intrinsic function
`strtod(3C)` — converts a string to a double-precision number

drawing

`grap(1)` — invokes a `pic` preprocessor for drawing graphs
`graph(1G)` — draws a graph
`pic(1)` — preprocesses `troff` files that contain drawings

drawings, generation of graphs and curves

`graph(1G)` — draws a graph
`spline(1G)` — interpolates a smooth curve

drawings, plotter, filtering for display purposes

`tplot(1G)` — interprets plotter instructions for use at a vintage display device

drivers

`console(7)` — provides access to the console keyboard and screen
`fd(7)` — provides an interface to 3.5-inch disk drives
`gd(7)` — provides a generic interface to disk devices
`intro(7)` — introduces device drivers and interfaces
`mouse(7)` — provides a mouse input device driver
`pty(7)` — provides a pseudo terminal driver
`serial(7)` — provides the on-board serial ports
`sxt(7)` — provides a pseudo-device driver
`tc(7)` — tape device driver

DTS 300 terminal

`300(1)` — filter text containing printer control sequences for a DASI terminal

duration

`time(1)` — prints the elapsed time during the execution of a command
`timex(1)` — reports the elapsed, user, and system time during the execution of a command

editors

`TextEditor(1)` — lets you edit files interactively through mouse and menu operations
`bfs(1)` — edits big files
`ed(1)` — edit text
`ex(1)` — edit text

nl(1) — processes a file through a line numbering filter
sed(1) — edits a stream of data
ssp(1) — produces single spaced output
vi(1) — invokes the screen-oriented (visual) display editor

effective group ID
getuid(2) — get real and effective user IDs and group IDs
setregid(2) — sets real and effective group ID

effective user ID
su(1) — substitutes user ID
getuid(2) — get real and effective user IDs and group IDs
setreuid(2) — set real and effective user ID
setsid(2P) — create session and set process group ID

emulation, terminal
CommandShell(1) — manages command-interpretation windows and moderates access to the A/UX console window
vt102(7) — provides protocols for VT102 terminals

enablers
enable(1) — enable or disable LP printers
mesg(1) — permits or denies the receipt of messages
accept(1M) — allows lp requests
acct(2) — enable or disable process accounting
phys(2) — allows a process to access physical addresses

encryption
crypt(1) — encodes and decodes passwords
makekey(1) — generates an encryption key
crypt(3C) — generate DES encryption

environment
env(1) — sets the environment for command execution
printenv(1) — displays the value of variables set in the current environment
getenv(3C) — return value for environment name
getenv(3F) — return Fortran environment variable
putenv(3C) — changes existing environmental variable values or adds new ones
profile(4) — setting up an environment at login time
environ(5) — user environment

error functions
erf(3M) — error function and complementary error function
matherr(3M) — provides an error-handling function

error logging
mkstr(1) — creates an error message file by massaging C source programs
errdemon(1M) — calls the error-logging daemon
errpt(1M) — processes a report of logged errors

errstop(1M) — terminates the error-logging daemon
errfile(4) — error-log file format
error(7) — interfaces between processes and error-record collection routines

errors, general

errdead(1M) — extracts error records from a crash dump
exterr(1M) — turns on/off the reporting of extended errors
intro(2) — introduces system calls and error numbers
matherr(3M) — provides an error-handling function
perror(3C) — produce system error messages

Ethernet

checkinstall(1) — checks the installation of boards
etheraddr(1M) — displays the Ethernet address of each Ethernet card in your system
ether(3N) — monitors Ethernet traffic
ethers(3N) — provide Ethernet address mapping operations
ethers(4) — Ethernet address to host name database or YP domain
ae(5) — 3Com 10 Mb/s Ethernet interface
arp(5P) — Address Resolution Protocol

Euclidean distance

hypot(3M) — provides the Euclidean distance function

evaluators

basename(1) — get part of a pathname
expr(1) — evaluates arguments as an expression
test(1) — evaluates conditions

execution, general

apply(1) — passes its arguments in batches to a command that is run once per every batch
at(1) — run commands at a later time
env(1) — sets the environment for command execution
launch(1) — runs a Macintosh binary application in A/UX
nice(1) — executes a command at low priority
nohup(1) — runs a command so that it can continue to run even after your session has ended
remsh(1N) — invokes to a shell on a remote system
sleep(1) — suspends the system for a specified interval of time
uux(1C) — runs a command on a remote system
xargs(1) — builds arguments based on the standard input, passing them in batches to the specified command which is executed enough times to deplete all the arguments
cron(1M) — runs the clock daemon
rexecd(1M) — server for remote executions
uuxqt(1M) — handles requests from remote systems to run commands
exec(2) — execute a file

regcmp(3X) — compile and execute a regular expression
 sleep(3C) — suspends execution for interval
 usleep(3) — suspend execution for interval

execution profile

prof(1) — displays profile data
 time(1) — prints the elapsed time during the execution of a command
 timex(1) — reports the elapsed, user, and system time during the
 execution of a command
 profil(2) — reports the execution time of an application
 monitor(3C) — prepares an execution profile

expanding and compressing files

compact(1) — compress and uncompress files
 compress(1) — compress files and directories as well as expand them;
 support concatenation, browsing, and file-comparing operations upon
 compressed files
 crypt(1) — encodes and decodes passwords
 makekey(1) — generates an encryption key
 pack(1) — compress and expand files

exponents

exp(3F) — Fortran exponential intrinsic function
 exp(3M) — provide exponential, logarithm, power, and square root
 functions

expressions

basename(1) — get part of a pathname
 expr(1) — evaluates arguments as an expression

expressions, regular

grep(1) — search a file for a specific pattern
 regcmp(1) — compiles regular expressions with a file
 regcmp(3X) — compile and execute a regular expression
 regexp(5) — regular expression compile and match routines

extended character set

greek(1) — filters text for vintage display devices

factoring

factor(1) — prints the prime factor of a given number

false and true

test(1) — evaluates conditions
 true(1) — provides truth values

fields

awk(1) — scans a file for lines that match a specific pattern
 colrm(1) — removes columns from a file
 cut(1) — cuts out selected fields of each line of a file
 join(1) — combines (joins) two relational files
 paste(1) — merges lines of several files or subsequent lines of one file
 sort(1) — sorts or merges files

qsort(3C) — performs a quicker sort

file control

touch(1) — updates access and modification times of a file

fcntl(2) — provides file control

fcntl(5) — file control options

file creation masks

umask(2) — set and get file creation mask

file formats used by A/UX

intro(4) — introduction to file formats

file handles

nfs_getfh(2) — gets a file handle

file handling

chmod(1) — changes the permissions of a file

chown(1) — change the owner or group of a file

cp(1) — copies files

cpio(1) — copies files to or from a cpio archive

csplit(1) — splits files into sections

dd(1) — converts and copies a file

fcvt(1) — converts a file in one storage format to a different storage format

file(1) — determines the type of a file

find(1) — finds files

head(1) — displays the first few lines of a file

ln(1) — makes links

lp(1) — spools print requests to printers

lpq(1) — queries the print spooler for progress information

lpr(1) — spools print requests to printers

lprm(1) — removes jobs from the line printer spooling queue for a

Berkeley file system (4.2)

ls(1) — lists the contents of a directory

mkdir(1) — creates a directory

more(1) — show the contents of a file in display-size chunks

mv(1) — moves or renames files

pax(1) — copies files to or from an archive in an IEEE format

pg(1) — shows the contents of a file in display-size chunks

rcp(1C) — copies files between two systems

rdist(1) — distributes remote files

rm(1) — remove files or directories

setfile(1) — sets attributes for Macintosh files, such as file type and creator

split(1) — splits a file into a specified number of pieces

sum(1) — calculates a checksum

tail(1) — displays the last part of a file

tar(1) — copies files to or from a tar archive

touch(1) — updates access and modification times of a file
 tp(1) — copies files to or from a tp archive
 updater(1) — updates files between two machines
 uusend(1C) — sends a file to a remote host
 uuto(1C) — provide an easy interface to the uucp command, using the
 public directories
 version(1) — reports version number of files
 clr(1M) — clears inodes
 ff(1M) — lists file names and statistics for a System V file system
 fuser(1M) — identifies processes using a file or file structure
 chown(2) — changes the owner and group of a file
 close(2) — closes a file descriptor
 creat(2) — creates a new file or rewrites an existing one
 exec(2) — execute a file
 link(2) — provides a link to a file
 lseek(2) — move read/write file pointer
 nfs_getfh(2) — gets a file handle
 open(2) — opens a file for reading or writing
 read(2) — reads from a file
 symlink(2) — make symbolic link to a file
 truncate(2) — truncate a file to a specified length
 write(2) — write on a file
 fopen(3S) — open a stream
 fread(3S) — produce binary input/output
 fseek(3S) — reposition a file pointer in a stream
 tmpfile(3S) — creates a temporary file
 fspec(4) — syntax for format lines for newform

file merging

cat(1) — catenates and displays the contents of files
 join(1) — combines (joins) two relational files
 merge(1) — merges three files into one
 paste(1) — merges lines of several files or subsequent lines of one file
 soelim(1) — eliminates the source commands from nroff input
 sort(1) — sorts or merges files
 tsort(1) — sorts lines in a file topologically
 acctmerge(1M) — merges or adds accounting files

file moving

mv(1) — moves or renames files

filenames

find(1) — finds files
 mv(1) — moves or renames files
 rename(2) — change the name of a file
 ctermid(3S) — generate filename for terminal
 mktemp(3C) — makes a unique filename

realpath(3) — returns the real filename of a file
tmpnam(3S) — create a name for a temporary file
fstypes(4) — name-mapping information for file systems

file permissions

chmod(1) — changes the permissions of a file
chown(1) — change the owner or group of a file
find(1) — finds files
ls(1) — lists the contents of a directory
chmod(2) — change mode of file
umask(2) — set and get file creation mask

file pointers

lseek(2) — move read/write file pointer
fseek(3S) — reposition a file pointer in a stream

file reading

cat(1) — catenates and displays the contents of files
head(1) — displays the first few lines of a file
line(1) — reads one line from the standard input
more(1) — show the contents of a file in display-size chunks
pg(1) — shows the contents of a file in display-size chunks
soelim(1) — eliminates the source commands from `nroff` input
tail(1) — displays the last part of a file
read(2) — reads from a file
fread(3S) — produce binary input/output
getc(3S) — get character or word from a stream

file regions

locking(2) — provides exclusive file regions for reading or writing
lockf(3C) — records locking on files

file scanning

cat(1) — catenates and displays the contents of files
head(1) — displays the first few lines of a file
line(1) — reads one line from the standard input
more(1) — show the contents of a file in display-size chunks
pg(1) — shows the contents of a file in display-size chunks
soelim(1) — eliminates the source commands from `nroff` input
tail(1) — displays the last part of a file
read(2) — reads from a file
fread(3S) — produce binary input/output
getc(3S) — get character or word from a stream

file status

chmod(1) — changes the permissions of a file
chown(1) — change the owner or group of a file
file(1) — determines the type of a file
find(1) — finds files
ls(1) — lists the contents of a directory

setfile(1) — sets attributes for Macintosh files, such as file type and creator
 sum(1) — calculates a checksum
 touch(1) — updates access and modification times of a file
 version(1) — reports version number of files
 ncheck(1M) — locates the filename associated with an i-number
 access(2) — determine accessibility of a file
 chmod(2) — change mode of file
 chown(2) — changes the owner and group of a file
 fsync(2) — synchronize a file's in-core state with that on disk
 stat(2) — get file status
 utime(2) — set file access and modification times
 stat(5) — data returned by stat system call

file system repair

clri(1M) — clears inodes
 fsck(1M) — checks file-system consistency and interactively repairs the file system
 fsdb(1M) — debugs the file system
 ncheck(1M) — locates the filename associated with an i-number
 esch(8) — validates and repairs file systems from the A/UX Startup shell

file systems, Berkeley

newfs(1M) — makes a Berkeley 4.2 (UFS) file system
 tuneufs(1M) — tunes a Berkeley 4.2 (UFS) file system
 ufs(4) — UFS file-system format

file systems, block zero information

bzb(4) — Block Zero Block file format

file systems, copying to backup media

bcopy(1M) — copies blocks interactively
 dcopy(1M) — copies System V File System-style file systems for optimal access time
 dump.bsd(1M) — create a dump.bsd archive by making copies of files from a given file system
 escher(1M) — helps you with autorecovery administration
 eu(1M) — updates autorecovery files
 eupdate(1M) — updates important files for autorecovery purposes
 finc(1M) — generates a fast incremental backup for System V file systems
 frec(1M) — recovers files from a backup tape
 restore(1M) — retrieve files from within a dump.bsd archive into an existing file system
 volcopy(1M) — copy file systems with label checking

file systems, display status of

- df(1) — reports the used and unused storage capacity for a file system
- du(1) — summarizes disk usage

file systems, general

- fstyp(1) — reports the file-system type
- sync(1) — updates the superblock
- automount(1M) — mounts Network File System (NFS) when needed
- clri(1M) — clears inodes
- devnm(1M) — displays the current device name
- exportfs(1M) — exports and unexports directories to Network File System (NFS) clients
- ff(1M) — lists file names and statistics for a System V file system
- fsck(1M) — checks file-system consistency and interactively repairs the file system
- fsdb(1M) — debugs the file system
- fsentry(1M) — creates an entry in the file-system table
- fsirand(1M) — installs random inode generation numbers
- fsstat(1M) — reports the state of a file system
- fuser(1M) — identifies processes using a file or file structure
- mkfs1b(1M) — constructs a file system with 512-byte blocks
- mkfs(1M) — constructs a System V file system
- mklost+found(1M) — makes a directory named `lost+found` to be used by `fsck`
- mount(1M) — mount and unmount file systems
- mountd(1M) — invokes the Network File System (NFS) mount-request server
- ncheck(1M) — locates the filename associated with an i-number
- newfs(1M) — makes a Berkeley 4.2 (UFS) file system
- tunefs(1M) — tunes a Berkeley 4.2 (UFS) file system
- volcopy(1M) — copy file systems with label checking
- fsmount(2) — mount a network file system (NFS)
- statfs(2) — gets file-system statistics
- umount(2) — unmount a file system
- unmount(2) — remove a file system
- ustat(2) — gets file system statistics
- exportent(3) — get exported file-system information
- fstyp(3) — determines the file-system type
- fstypent(3P) — gets a file-system-type entry
- ftw(3C) — walks a file tree
- getmntent(3) — get file system descriptor file entry
- mount(3) — mounts a file system
- mount(3N) — keeps track of remotely mounted file systems
- umount(3) — unmounts a file system
- dump.bsd(4) — format of a file-system dump

`exports(4)` — directories to export to Network File System (NFS) clients
`fs(4)` — file systems
`fstab(4)` — parameter file format
`fstypes(4)` — name-mapping information for file systems
`mtab(4)` — mounted file system table
`rmtab(4)` — remotely mounted file system table
`svfs(4)` — System V system volume format
`ufs(4)` — UFS file-system format
`esch(8)` — validates and repairs file systems from the A/UX Startup shell

file systems, maintenance

`fstyp(1)` — reports the file-system type
`sync(1)` — updates the superblock
`clri(1M)` — clears inodes
`devnm(1M)` — displays the current device name
`ff(1M)` — lists file names and statistics for a System V file system
`fsck(1M)` — checks file-system consistency and interactively repairs the file system
`fsdb(1M)` — debugs the file system
`fsentry(1M)` — creates an entry in the file-system table
`fsirand(1M)` — installs random inode generation numbers
`fsstat(1M)` — reports the state of a file system
`fuser(1M)` — identifies processes using a file or file structure
`mkfs1b(1M)` — constructs a file system with 512-byte blocks
`mkfs(1M)` — constructs a System V file system
`mklost+found(1M)` — makes a directory named `lost+found` to be used by `fsck`
`mount(1M)` — mount and unmount file systems
`ncheck(1M)` — locates the filename associated with an i-number
`newfs(1M)` — makes a Berkeley 4.2 (UFS) file system
`tunefs(1M)` — tunes a Berkeley 4.2 (UFS) file system

file systems, NFS, maintenance of

`domainname(1)` — sets or displays the name of the Network Information Service (NIS) domain
`automount(1M)` — mounts Network File System (NFS) when needed
`exportfs(1M)` — exports and unexports directories to Network File System (NFS) clients
`lockd(1M)` — handle local and remote lock requests
`mountd(1M)` — invokes the Network File System (NFS) mount-request server
`nfsd(1M)` — invoke the NFS daemons
`nfsstat(1M)` — displays Network File System (NFS) statistics
`rpcinfo(1M)` — reports RPC information
`showmount(1M)` — shows all remote mounts
`spray(1M)` — sprays packets

sprayd(1M) — returns information for the `spray` command
statd(1M) — provide crash and recovery monitoring for network locking services

file systems, System V

mkfs(1M) — constructs a System V file system
dir(4) — format of System V directories
inode(4) — format of a System V inode
svfs(4) — System V system volume format

file systems, unmounting

umount(2) — unmount a file system
umount(2) — remove a file system
umount(3) — unmounts a file system

file transfers

cpio(1) — copies files to or from a `cpio` archive
cu(1C) — establishes an interactive connection with another system
ftp(1N) — transfers files by using the DARPA Internet File Transfer Protocol (FTP)
kermit(1C) — invokes the Kermit file-transfer program
pax(1) — copies files to or from an archive in an IEEE format
rcp(1C) — copies files between two systems
remsh(1N) — invokes to a shell on a remote system
tar(1) — copies files to or from a `tar` archive
tftp(1C) — transfers files via the Trivial File Transfer Protocol (TFTP)
tip(1C) — establishes a connection to a remote system
updater(1) — updates files between two machines
uucp(1C) — copies files from one system to another system
uuencode(1C) — encode and decode a binary file
ftpd(1M) — provide Internet File Transfer Protocol (FTP) service
tftpd(1M) — responds to requests to use the DARPA Trivial File Transfer Protocol
uucico(1M) — transfers files as specified by `uucp` work files

file types

file(1) — determines the type of a file
find(1) — finds files
magic(4) — magic number file for `file` command

file writing

write(2) — write on a file

files, archive

ar(1) — maintains a library of files in an archive
cpio(1) — copies files to or from a `cpio` archive
lorder(1) — finds the ordering relation for an object library
pax(1) — copies files to or from an archive in an IEEE format
tar(1) — copies files to or from a `tar` archive
ldahread(3X) — reads the archive header of a member of an archive file

- ar(4) — common archive file format
- cpio(4) — format of cpio archive
- tar(4) — format of tar header

files, big

- bdiff(1) — compares the difference between two large files that are too big for diff to handle
- bfs(1) — edits big files

files, browsing

- head(1) — displays the first few lines of a file
- more(1) — show the contents of a file in display-size chunks
- pg(1) — shows the contents of a file in display-size chunks
- tail(1) — displays the last part of a file

files, comparing

- bdiff(1) — compares the difference between two large files that are too big for diff to handle
- cmp(1) — compares two files
- comm(1) — selects or rejects lines common to two sorted files
- diff3(1) — compares three versions of a file
- diff(1) — compares two files or directories for any differences
- dircmp(1) — compares the contents of two directories
- merge(1) — merges three files into one
- rcsdiff(1) — compares RCS revisions
- sccsdiff(1) — compares two versions of an SCCS file
- sdiff(1) — reports side-by-side differences between two files in a side-by-side format
- sumdir(1) — sums and counts the characters within the files of the given directories
- ucbdiff3(1) — reports the differences between three files
- ucbdiff(1) — reports differences between two files or directories
- uniq(1) — reports repeated lines in a file

files, compressing and expanding

- compact(1) — compress and uncompress files
- compress(1) — compress files and directories as well as expand them; support concatenation, browsing, and file-comparing operations upon compressed files
- crypt(1) — encodes and decodes passwords
- makekey(1) — generates an encryption key
- pack(1) — compress and expand files

files, copying

- atprint(1) — transfers data to a printer by using AppleTalk protocols
- cp(1) — copies files
- cpio(1) — copies files to or from a cpio archive
- csplit(1) — splits files into sections
- dd(1) — converts and copies a file

fcnv(1) — converts a file in one storage format to a different storage format
ln(1) — makes links
pax(1) — copies files to or from an archive in an IEEE format
rcp(1C) — copies files between two systems
split(1) — splits a file into a specified number of pieces
tar(1) — copies files to or from a tar archive
tp(1) — copies files to or from a tp archive
uucp(1C) — copies files from one system to another system
uuto(1C) — provide an easy interface to the uucp command, using the public directories
bcopy(1M) — copies blocks interactively
dcopy(1M) — copies System V File System-style file systems for optimal access time
dump.bsd(1M) — create a dump.bsd archive by making copies of files from a given file system
restore(1M) — retrieve files from within a dump.bsd archive into an existing file system
volcopy(1M) — copy file systems with label checking
blt(3C) — block transfer data
cpio(4) — format of cpio archive
tar(4) — format of tar header
files, device description
printcap(4) — printer-capability database
termcap(4) — terminal capability database
terminfo(4) — terminal capability database
files, displaying status of
file(1) — determines the type of a file
ls(1) — lists the contents of a directory
sum(1) — calculates a checksum
version(1) — reports version number of files
files, dividing
csplit(1) — splits files into sections
split(1) — splits a file into a specified number of pieces
files, FIFO
mkfifo(3P) — makes a FIFO special file
files, finding
find(1) — finds files
files, manipulating
cp(1) — copies files
cpio(1) — copies files to or from a cpio archive
csplit(1) — splits files into sections
dd(1) — converts and copies a file
fcnv(1) — converts a file in one storage format to a different storage

format

`ln(1)` — makes links
`mkdir(1)` — creates a directory
`mv(1)` — moves or renames files
`pax(1)` — copies files to or from an archive in an IEEE format
`rcp(1C)` — copies files between two systems
`rm(1)` — remove files or directories
`split(1)` — splits a file into a specified number of pieces
`tar(1)` — copies files to or from a `tar` archive
`tp(1)` — copies files to or from a `tp` archive

files, merging

`cat(1)` — catenates and displays the contents of files
`join(1)` — combines (joins) two relational files
`merge(1)` — merges three files into one
`paste(1)` — merges lines of several files or subsequent lines of one file
`soelim(1)` — eliminates the source commands from `nroff` input
`sort(1)` — sorts or merges files
`tsort(1)` — sorts lines in a file topologically
`acctmerg(1M)` — merges or adds accounting files

files, Name Information Server

`makedbm(1M)` — generates a Network Information Service (NIS) `dbm` file

files, printing

`cancel(1)` — cancels print requests spooled through the `lp` command
`lp(1)` — spools print requests to printers
`lpq(1)` — queries the print spooler for progress information
`lpr(1)` — spools print requests to printers
`lprm(1)` — removes jobs from the line printer spooling queue for a Berkeley file system (4.2)

files, RCS

`ci(1)` — checks in RCS revisions
`co(1)` — checks out RCS revisions
`ident(1)` — displays RCS keywords and their values
`merge(1)` — merges three files into one
`rccs(1)` — creates new RCS files or changes attributes of existing RCS files
`rcsdiff(1)` — compares RCS revisions
`rcsintro(1)` — introduces RCS commands
`rcsmmerge(1)` — merges two versions of an RCS file
`rlog(1)` — displays log messages and other information about RCS files
`ucbdiff3(1)` — reports the differences between three files
`ucbdiff(1)` — reports differences between two files or directories
`sccstorcs(1M)` — builds an RCS file from an SCCS file
`rcsfile(4)` — format of an RCS file

files, SCCS

`admin(1)` — creates and administers SCCS files
`cdc(1)` — changes the delta commentary of an SCCS delta
`comb(1)` — combines SCCS deltas
`delta(1)` — makes a delta (change) to an SCCS file
`get(1)` — gets a version of an SCCS file
`help(1)` — provides help information about SCCS commands and messages
`prs(1)` — displays information about an SCCS file
`rmdel(1)` — removes a delta from an SCCS file
`sact(1)` — displays who has checked a Source Code Control System (SCCS) file out for editing
`sccs(1)` — performs SCCS subsystem commands
`sccsdiff(1)` — compares two versions of an SCCS file
`unget(1)` — undoes a previous get of an SCCS file
`val(1)` — validate SCCS file
`what(1)` — reports identification information for a file
`sccstorcs(1M)` — builds an RCS file from an SCCS file
`sccsfile(4)` — format of an SCCS file

files, searching for

`find(1)` — finds files

finding files

`find(1)` — finds files

flag options

`getopt(1)` — parses command options
`getopt(3C)` — get option letter from argument vector

floating-point numbers

`atof(3C)` — converts an ASCII string to floating-point number
`ecvt(3C)` — convert floating-point number to string
`frexp(3C)` — manipulate parts of floating-point numbers

floor numbers

`floor(3M)` — floor, ceiling, remainder, absolute value functions

floppy disks

`cpio(1)` — copies files to or from a `cpio` archive
`eject(1)` — ejects a diskette from the drive
`pax(1)` — copies files to or from an archive in an IEEE format
`tar(1)` — copies files to or from a `tar` archive
`diskformat(1M)` — formats a disk through a driver-dependent format operation
`finstall(1M)` — installs A/UX software from specially prepared floppy disks
`cpio(4)` — format of `cpio` archive
`finstallrc(4)` — `finstall` default configuration file
`tar(4)` — format of `tar` header

fd(7) — provides an interface to 3.5-inch disk drives

flowgraphs

cflow(1) — generates a C flowgraph

font files, troff

makedev(1) — prepares troff description files

afm(4) — Adobe POSTSCRIPT font metrics file format

font(5) — description files for device-independent troff

footnotes

mm(1) — formats documents that contain nroff and mm macro formatting requests

refer(1) — finds and inserts literature references in documents

me(5) — macros for formatting papers

mm(5) — macro package for formatting documents

ms(5) — text formatting macros

format checkers

checkmm(1) — check documents formatted with the mm macros

checknr(1) — checks nroff/troff files

lint(1) — invokes a C program checker

format macros

checkmm(1) — check documents formatted with the mm macros

m4(1) — processes macros for C and other languages

macref(1) — produces a cross-reference listing of macro files

mm(1) — formats documents that contain nroff and mm macro formatting requests

man(5) — macros for formatting entries in this manual

me(5) — macros for formatting papers

mm(5) — macro package for formatting documents

mptx(5) — the macro package for formatting a permuted index

ms(5) — text formatting macros

mv(5) — a troff macro package for typesetting viewgraphs and slides

formatters, disk

diskformat(1M) — formats a disk through a driver-dependent format operation

formatters, text

daps(1) — invokes the Autologic APS-5 phototypesetter troff post-processor

enscript(1) — converts text files to format for printing

eqn(1) — format mathematical text for troff

fmt(1) — invokes a simple text formatter

fold(1) — folds long lines for finite-width output device

mm(1) — formats documents that contain nroff and mm macro formatting requests

mmt(1) — typeset documents that contain troff and mm or mv macro-formatting requests

mvt(1) — typeset documents that contain troff and mm or mv macro-formatting requests
neqn(1) — formats mathematical text for nroff
newform(1) — changes the format of a text file
nroff(1) — text formatter
otroff(1) — formats text for a specific phototypesetter
pr(1) — formats text for a print device
psdit(1) — converts troff intermediate format to POSTSCRIPT format
psroff(1) — formats a file through troff so it can be printed on a POSTSCRIPT printer
roffbib(1) — prints out all records in a bibliographic database
tbl(1) — table formatter for nroff or troff
troff(1) — formats and typesets files

Fortran facilities

asa(1) — interprets ASA carriage control characters
efl(1) — invokes the Extended Fortran Language
f77(1) — invokes the Fortran 77 compiler
fpr(1) — filters the output of Fortran programs for line printing
fsplit(1) — splits f77 or efl files
abort(3F) — terminates a Fortran program
abs(3F) — Fortran absolute value
acos(3F) — Fortran arccosine intrinsic function
aimag(3F) — Fortran imaginary part of complex argument
aint(3F) — Fortran integer part intrinsic function
asin(3F) — Fortran arcsine intrinsic function
atan2(3F) — Fortran arctangent intrinsic function
atan(3F) — Fortran arctangent intrinsic function
bool(3F) — Fortran bitwise boolean functions
conjg(3F) — Fortran complex conjugate intrinsic function
cos(3F) — Fortran cosine intrinsic function
cosh(3F) — Fortran hyperbolic cosine intrinsic function
dim(3F) — Fortran positive difference intrinsic functions
dprod(3F) — Fortran double precision product intrinsic function
exp(3F) — Fortran exponential intrinsic function
ftype(3F) — explicit Fortran type conversion
getarg(3F) — return Fortran command-line argument
getenv(3F) — return Fortran environment variable
iargc(3F) — return command line arguments
index(3F) — return location of Fortran substring
len(3F) — return length of Fortran string
lge(3F) — string comparison intrinsic functions
log10(3F) — Fortran common logarithm intrinsic function
log(3F) — Fortran natural logarithm intrinsic function
max(3F) — provides Fortran maximum-value functions

mclock(3F) — returns Fortran time accounting
min(3F) — provide Fortran minimum-value functions
mod(3F) — provide Fortran remaindering intrinsic functions
rand(3F) — provide a Fortran uniform random-number generator
round(3F) — provide Fortran nearest integer functions
sign(3F) — returns Fortran transfer-of-sign intrinsic functions
signal(3F) — specifies Fortran action on receipt of a system signal
sin(3F) — provide Fortran sine intrinsic functions
sinh(3F) — provide Fortran hyperbolic sine intrinsic function
sqrt(3F) — provide Fortran square root intrinsic functions
system(3F) — issues a shell command from Fortran
tan(3F) — Fortran tangent intrinsic function
tanh(3F) — Fortran hyperbolic tangent intrinsic function

Fortran programming

asa(1) — interprets ASA carriage control characters
efl(1) — invokes the Extended Fortran Language
f77(1) — invokes the Fortran 77 compiler
fpr(1) — filters the output of Fortran programs for line printing
fsplit(1) — splits f77 or efl files

full-duplex

shutdown(2N) — shut down part of a full-duplex connection
termio(7) — provides a general terminal interface
termios(7P) — provides a A/UX® POSIX general terminal interface

games

aliens(6) — plays the game of Space Invaders (A/UX version)
arithmetic(6) — provides arithmetic problems
autorobots(6) — plays the game of autorobots
back(6) — plays the game of backgammon
bcd(6) — simulates a punched card corresponding to a text argument
bj(6) — plays the game of black jack
chase(6) — plays the game of chase
craps(6) — plays the game of craps
cribbage(6) — plays the game of cribbage
fish(6) — plays the game of Go Fish''
fortune(6) — plays the game of fortune telling
hangman(6) — plays the game of hangman
intro(6) — introduction to games
life(6) — plays the game of life
mastermind(6) — plays the game of Mastermind
maze(6) — generates a maze
moo(6) — plays the game of moo
number(6) — converts Arabic numerals to English
quiz(6) — gives associative knowledge tests on various subjects
rain(6) — animates raindrops

- robots(6) — plays the game of robots
- trek(6) — plays the game of trek
- ttt(6) — play the game of tic-tac-toe
- twinkle(6) — plays the game of twinkle, twinkle little stars
- worm(6) — plays the game of growing worm
- worms(6) — plays the game of worms
- wump(6) — plays the game of hunt-the-wumpus
- gamma function**
 - gamma(3M) — logs a gamma function
- geometry**
 - hypot(3M) — provides the Euclidean distance function
- go fish**
 - fish(6) — plays the game of Go Fish''
- goto**
 - setjmp(3C) — provide non-local goto
 - sigsetjmp(3P) — provide non-local jumps
- graphics**
 - graph(1G) — draws a graph
 - pic(1) — preprocesses troff files that contain drawings
 - spline(1G) — interpolates a smooth curve
 - tplot(1G) — interprets plotter instructions for use at a vintage display device
 - plot(3X) — provide graphics interface subroutines
 - plot(4) — graphics interface
- graphs**
 - grap(1) — invokes a pic preprocessor for drawing graphs
 - graph(1G) — draws a graph
- group access lists**
 - getgroups(2) — gets group access list
 - setgroups(2) — sets group access list
 - initgroups(3) — initializes group access list
- group IDs**
 - id(1) — displays user and group IDs and names
 - setuid(2) — set user and group ID
 - group(4) — group file format
 - passwd(4) — password file
- groups**
 - chown(1) — change the owner or group of a file
 - groups(1) — displays group memberships
 - id(1) — displays user and group IDs and names
 - newgrp(1) — logs you into a new group
 - pwck(1M) — check the password/group files
 - chown(2) — changes the owner and group of a file
 - getgroups(2) — gets group access list

getuid(2) — get real and effective user IDs and group IDs
setgroups(2) — sets group access list
setregid(2) — sets real and effective group ID
setuid(2) — set user and group ID
getgrent(3C) — obtain group file entry from a group file
initgroups(3) — initializes group access list
group(4) — group file format

GSI 300 terminal

300(1) — filter text containing printer control sequences for a DASI terminal

half-duplex

shutdown(2N) — shut down part of a full-duplex connection
termio(7) — provides a general terminal interface
termios(7P) — provides a A/UX® POSIX general terminal interface

halting execution

kill(1) — terminates a process
killall(1M) — kills all active processes
exit(2) — terminate process
kill(2) — sends a signal to a process or a group of processes
reboot(2) — reboot system or halt processor

handle, file

nfs_getfh(2) — gets a file handle

hangman

hangman(6) — plays the game of hangman

hash tables

hsearch(3C) — manage hash search tables

help, command options

cmdo(1) — builds command lines interactively

help, online

apropos(1) — locates commands by keyword
man(1) — displays the named manual page entries
whatis(1) — reports a brief description for the manual page entry specified
whereis(1) — reports the locations of the source, binary, and online help files for a specified command
which(1) — reports the directory path to a file by interpreting PATH and alias settings

host names

HOSTNAME(4) — host name and domain name database
ethers(4) — Ethernet address to host name database or YP domain
hosts.equiv(4) — files containing a list of trusted hosts
hosts(4) — host name database

hosts

hostid(1N) — sets or displays the identifier of the current host system
hostname(1N) — sets or displays the name of the current host system
uname(1) — displays identification information about the current system
uname(2) — get name of current system
byteorder(3N) — convert values between host and network byte order
gethostbyaddr(3N) — get network host entry
hosts.equiv(4) — files containing a list of trusted hosts
hosts(4) — host name database
remote(4) — remote host description file
rhosts(4N) — trusted hosts file format
slip.hosts(4) — maps login names to Compressed Serial Line/Internet
 Protocol (CSL/IP) client host names

HUGE (constant)

math(5) — math functions and constants

hyperbolic functions

cosh(3F) — Fortran hyperbolic cosine intrinsic function
sinh(3F) — provide Fortran hyperbolic sine intrinsic function
sinh(3M) — provide hyperbolic functions
tanh(3F) — Fortran hyperbolic tangent intrinsic function

hyphenation

hyphen(1) — finds hyphenated words

I/O management

query(1) — queries the user for input
tee(1) — transcribes data
ioctl(2) — requests low-level, input/output operations for specific
 devices
select(2N) — synchronous I/O multiplexing
cfgetospeed(3P) — get or set the value of the output and input baud
 rate
fread(3S) — produce binary input/output
fseek(3S) — reposition a file pointer in a stream
printf(3S) — format and output string and numeric data
scanf(3S) — convert formatted input
streams(7) — provides an interface for character I/O

ICMP

icmp(5P) — Internet Control Message Protocol

IDs

id(1) — displays user and group IDs and names
setuid(2) — set user and group ID
auxstartuprc(4) — authorization file that helps password-protect and
 otherwise secure A/UX Startup
group(4) — group file format
passwd(4) — password file

ImageWriter

iw2(1) — prepares data to be printed on the Apple ImageWriter II printer

indexing

indxbib(1) — builds an inverted index for a bibliography

ndx(1) — creates a subject-page index for a document

ptx(1) — generates a permuted index

initialization

tset(1) — set or reset the terminal to a sensible state

brc(1M) — execute system initialization shell scripts

init(1M) — spawn general processes

inittab(4) — script for the **init** process

inittab file

init(1M) — spawn general processes

tty_add(1M) — modify the **/etc/inittab** file in terms of enabling serial ports for use as login terminals

inittab(4) — script for the **init** process

inodes

clri(1M) — clears inodes

fsck(1M) — checks file-system consistency and interactively repairs the file system

fsirand(1M) — installs random inode generation numbers

mkfs(1M) — constructs a System V file system

ncheck(1M) — locates the filename associated with an i-number

newfs(1M) — makes a Berkeley 4.2 (UFS) file system

inode(4) — format of a System V inode

Input/Output management

query(1) — queries the user for input

tee(1) — transcribes data

ioctl(2) — requests low-level, input/output operations for specific devices

select(2N) — synchronous I/O multiplexing

cfgetospeed(3P) — get or set the value of the output and input baud rate

fread(3S) — produce binary input/output

fseek(3S) — reposition a file pointer in a stream

printf(3S) — format and output string and numeric data

scanf(3S) — convert formatted input

soundinput(7) — provides interface conventions for the sound input driver

streams(7) — provides an interface for character I/O

vt102(7) — provides protocols for VT102 terminals

installers

- `cpset(1M)` — installs files in specified directories
- `finstall(1M)` — installs A/UX software from specially prepared floppy disks
- `fsirand(1M)` — installs random inode generation numbers
- `install(1M)` — places files in specified directories
- `mklost+found(1M)` — makes a directory named `lost+found` to be used by `fsck`
- `ypinit(1M)` — initializes Network Information Service (NIS) maps for master and slave servers
- `finstallrc(4)` — `finstall` default configuration file

integers

- `bc(1)` — processes an arbitrary-precision arithmetic language
- `dc(1)` — desk calculator
- `expr(1)` — evaluates arguments as an expression
- `factor(1)` — prints the prime factor of a given number
- `abs(3C)` — return integer absolute value
- `abs(3F)` — Fortran absolute value
- `aint(3F)` — Fortran integer part intrinsic function
- `drand48(3C)` — generate uniformly distributed pseudo-random numbers
- `rand(3C)` — call a simple random-number generator
- `rand(3F)` — provide a Fortran uniform random-number generator
- `round(3F)` — provide Fortran nearest integer functions
- `strtol(3C)` — convert strings to integer

interfaces

- `telnet(1C)` — communicates with another host via the TELNET protocol
- `appletalk(1M)` — enables you to configure and display AppleTalk network interfaces
- `ifconfig(1M)` — manages network interfaces
- `atp(3N)` — provide a AppleTalk Transaction Protocol (ATP) interface
- `ddp(3N)` — provide an AppleTalk Datagram Delivery Protocol (DDP) interface
- `lap(3N)` — AppleTalk Link Access Protocol (LLAP/ELAP) interface
- `nbp(3N)` — perform AppleTalk Name Binding Protocol (NBP) interface operations
- `pap(3N)` — provide AppleTalk Printer Access Protocol (PAP) interface
- `plot(3X)` — provide graphics interface subroutines
- `set42sig(3)` — sets the Berkeley Software Distribution (BSD) 4.2 signal interface
- `ypclnt(3N)` — provide a Network Information Service (NIS) client interface
- `zip(3N)` — provide a AppleTalk Zone Information Protocol (ZIP) interface

plot(4) — graphics interface
 slip.config(4) — establishes the number of available Compressed
 Serial Line/Internet Protocol (CSL/IP) connections
 ae(5) — 3Com 10 Mb/s Ethernet interface
 lo(5) — software loopback network interface
 appletalk(7) — interfaces with the AppleTalk protocols
 error(7) — interfaces between processes and error-record collection
 routines
 gd(7) — provides a generic interface to disk devices
 intro(7) — introduces device drivers and interfaces
 mem(7) — provide an interface for access to core memory
 mtio(7) — provides an interface library for magnetic tape devices
 nvram(7) — provides an interface to nonvolatile memory
 streams(7) — provides an interface for character I/O
 termio(7) — provides a general terminal interface
 termios(7P) — provides a A/UX® POSIX general terminal interface
 tty(7) — controls the terminal interface

Internet Control Message Protocol

 icmp(5P) — Internet Control Message Protocol

Internet, general

 ftp(1N) — transfers files by using the DARPA Internet File Transfer
 Protocol (FTP)
 nslookup(1) — interactively queries name servers
 rmail(1) — handles remote mail received via UUCP
 tftp(1C) — transfers files via the Trivial File Transfer Protocol (TFTP)
 ftpd(1M) — provide Internet File Transfer Protocol (FTP) service
 inetd(1M) — starts Internet servers when needed
 named(1M) — provides Internet domain name service
 portmap(1M) — converts RPC program numbers into DARPA protocol
 port numbers
 sendmail(1M) — sends mail
 stdhosts(1M) — converts Internet addresses to standard form
 telnetd(1M) — supports the DARPA standard TELNET protocol
 tftpd(1M) — responds to requests to use the DARPA Trivial File
 Transfer Protocol
 inet(3N) — provide Internet address manipulation routines
 resolver(3N) — provide resolver routines
 networks(4N) — network name database
 protocols(4N) — protocol name database
 resolv.conf(4) — configuration file for resolver routines
 servers(4) — Internet server database
 services(4N) — service name database
 arp(5P) — Address Resolution Protocol
 icmp(5P) — Internet Control Message Protocol

inet(5P) — Internet protocol family
ip(5P) — Internet Protocol
tcp(5P) — Internet Transmission Control Protocol
udp(5P) — Internet User Datagram Protocol

interpolator

soelim(1) — eliminates the source commands from nroff input
spline(1G) — interpolates a smooth curve

interpreters

bs(1) — compiles and interprets bs programs
csh(1) — runs the C shell, a command interpreter with C-like syntax
ksh(1) — runs the Korn shell, an enhanced command interpreter that is
backward-compatible with the Bourne shell (sh)
sh(1) — runs the Bourne shell
sno(1) — runs the SNOBOL interpreter
StartupShell(8) — interprets command lines such as those used to
boot A/UX and check file systems within the A/UX Startup
application

Interpreting commands

csh(1) — runs the C shell, a command interpreter with C-like syntax
ksh(1) — runs the Korn shell, an enhanced command interpreter that is
backward-compatible with the Bourne shell (sh)
sh(1) — runs the Bourne shell

interprocess communication

ipcrm(1) — removes interprocess communications facilities
ipcs(1) — reports interprocess communication facilities status
kill(1) — terminates a process
msgctl(2) — message control operations
msgget(2) — gets message queue
msgop(2) — message operations
semctl(2) — semaphore control operations
semget(2) — get set of semaphores
semop(2) — performs semaphore operations
shmctl(2) — shared memory control operations
shmget(2) — get shared memory segment
shmop(2) — shared memory operations
ftok(3C) — standard interprocess communication package

interval timers

getitimer(2) — get/set value of interval timer

IOT faults

abort(3C) — generates an IOT fault

ISO encoding

mactois(1) — convert between Macintosh encoding and International
Standards Organization (ISO) encoding

issue

issue(4) — project identification file format

job control

at(1) — run commands at a later time

crontab(1) — aids in the use of the cron process scheduling program

csh(1) — runs the C shell, a command interpreter with C-like syntax

env(1) — sets the environment for command execution

ksh(1) — runs the Korn shell, an enhanced command interpreter that is backward-compatible with the Bourne shell (sh)

nice(1) — executes a command at low priority

nohup(1) — runs a command so that it can continue to run even after your session has ended

sh(1) — manages the layering of multiple shells

sleep(1) — suspends the system for a specified interval of time

yes(1) — generates y entries in response to requests for input

chroot(1M) — changes the root directory for a command

cron(1M) — runs the clock daemon

join files relationally

join(1) — combines (joins) two relational files

Kermit

kermit(1C) — invokes the Kermit file-transfer program

kernels

uname(1) — displays identification information about the current system

autoconfig(1M) — creates an up-to-date kernel

chgnod(1M) — changes the current A/UX system node name

kconfig(1M) — tunes kernel parameters for work-load optimization

module_dump(1M) — queries kernel files for configuration information

ncstats(1M) — displays kernel name cache statistics

newconfig(1M) — generates an up-to-date kernel

newunix(1M) — manipulates the files that determine the configuration of a new kernel

rstatd(1M) — invokes a server for kernel statistics

uvar(2) — returns system-specific configuration information

rstat(3N) — get performance data from remote kernel

master(4) — master kernel-configuration file format

mem(7) — provide an interface for access to core memory

launch(8) — launches an A/UX kernel from the A/UX Startup environment

keyboard maps

keyset(1M) — sets the keyboard for the console

keys (encryption)

crypt(1) — encodes and decodes passwords

makekey(1) — generates an encryption key

crypt(3C) — generate DES encryption

keywords

`apropos(1)` — locates commands by keyword
`ident(1)` — displays RCS keywords and their values
`ndx(1)` — creates a subject-page index for a document
`subj(1)` — generates a list of subjects from documents
`ypmatch(1)` — lists the value of a specified key in a Network Information Service (NIS) map

Korn shell

`ksh(1)` — runs the Korn shell, an enhanced command interpreter that is backward-compatible with the Bourne shell (`sh`)

labels

`volcopy(1M)` — copy file systems with label checking

languages

`awk(1)` — scans a file for lines that match a specific pattern
`bc(1)` — processes an arbitrary-precision arithmetic language
`bs(1)` — compiles and interprets `bs` programs
`cc(1)` — invokes the C compiler
`cpp(1)` — invokes the C language preprocessor
`csh(1)` — runs the C shell, a command interpreter with C-like syntax
`efl(1)` — invokes the Extended Fortran Language
`eqn(1)` — format mathematical text for `troff`
`f77(1)` — invokes the Fortran 77 compiler
`ksh(1)` — runs the Korn shell, an enhanced command interpreter that is backward-compatible with the Bourne shell (`sh`)
`lex(1)` — generates programs for simple lexical tasks
`neqn(1)` — formats mathematical text for `nroff`
`nroff(1)` — text formatter
`pic(1)` — preprocesses `troff` files that contain drawings
`rpcgen(1)` — generates C source code from a remote procedure call (RPC) source file
`sh(1)` — runs the Bourne shell
`sno(1)` — runs the SNOBOL interpreter
`tbl(1)` — table formatter for `nroff` or `troff`
`troff(1)` — formats and typesets files
`yacc(1)` — compiles compilers (yet another compiler-compiler)

LAP

`lap(3N)` — AppleTalk Link Access Protocol (LLAP/ELAP) interface

launching Macintosh applications from the command line

`launch(1)` — runs a Macintosh binary application in A/UX

lexical analysis

`awk(1)` — scans a file for lines that match a specific pattern
`lex(1)` — generates programs for simple lexical tasks

library management

`ar(1)` — maintains a library of files in an archive
`mkshlib(1)` — creates a shared library

life

`life(6)` — plays the game of life

line counting

`wc(1)` — counts characters, words, and lines in a file

line discipline

`stty(1)` — sets the modes for a terminal
`line_sane(1M)` — pushes streams line disciplines
`line_push(3)` — routine used to push streams line disciplines
`termio(7)` — provides a general terminal interface

line numbering

`ld(1)` — invokes the link editor for common object files
`nl(1)` — processes a file through a line numbering filter
`pr(1)` — formats text for a print device
`strip(1)` — strips symbol and line number information from an object file
`linenum(4)` — line number entries in a common object file

lines, blank (in text)

`ssp(1)` — produces single spaced output

lines, filling and wrapping

`fmt(1)` — invokes a simple text formatter
`fold(1)` — folds long lines for finite-width output device

lines, processing text within

`awk(1)` — scans a file for lines that match a specific pattern
`colrm(1)` — removes columns from a file
`comm(1)` — selects or rejects lines common to two sorted files
`cut(1)` — cuts out selected fields of each line of a file
`grep(1)` — search a file for a specific pattern
`head(1)` — displays the first few lines of a file
`join(1)` — combines (joins) two relational files
`line(1)` — reads one line from the standard input
`newform(1)` — changes the format of a text file
`nl(1)` — processes a file through a line numbering filter
`paste(1)` — merges lines of several files or subsequent lines of one file
`rev(1)` — reverses characters within each line of text
`sed(1)` — edits a stream of data
`sort(1)` — sorts or merges files
`tail(1)` — displays the last part of a file
`uniq(1)` — reports repeated lines in a file
`wc(1)` — counts characters, words, and lines in a file

lines, repeated (in text)

`uniq(1)` — reports repeated lines in a file

lines, reversing characters within

`rev(1)` — reverses characters within each line of text

Link Access Protocol

`lap(3N)` — AppleTalk Link Access Protocol (LLAP/ELAP) interface

link editor (object code)

`ld(1)` — invokes the link editor for common object files

`a.out(4)` — common assembler and link editor output

links, file

`ln(1)` — makes links

`link(2)` — provides a link to a file

`readlink(2)` — read value of a symbolic link

`symlink(2)` — make symbolic link to a file

listening

`listen(2N)` — listens for connections on a socket

literary style

`diction(1)` — locate wordy sentences in a document

`spell(1)` — find spelling errors

`style(1)` — analyzes the surface characteristics of documents

locking

`locking(2)` — provides exclusive file regions for reading or writing

`plock(2)` — enables a lock process for text or data in memory

`lockf(3C)` — records locking on files

logarithms

`exp(3F)` — Fortran exponential intrinsic function

`exp(3M)` — provide exponential, logarithm, power, and square root functions

`log10(3F)` — Fortran common logarithm intrinsic function

`log(3F)` — Fortran natural logarithm intrinsic function

`math(5)` — math functions and constants

logging in and logging out

`login(1)` — signs you on a terminal session

`logname(1)` — gets the login name

`newgrp(1)` — logs you into a new group

`passwd(1)` — changes the login password

`rlogin(1N)` — logs in to a remote system

`Login(1M)` — logs you in to A/UX by using a graphical user interface

`remlogin(1M)` — runs on a remote system to log you in

`rlogind(1M)` — server for remote logins

`getlogin(3C)` — gets login name

`getusershell(3)` — generate authenticated pathnames corresponding to executable shell programs

`logname(3X)` — return login name of user

auxstartuprc(4) — authorization file that helps password-protect and otherwise secure A/UX Startup
 issue(4) — project identification file format
 passwd(4) — password file
 profile(4) — setting up an environment at login time
 shells(4) — shell pathnames file

long integers

a64l(3C) — convert between long integer and base-64 ASCII string
 drand48(3C) — generate uniformly distributed pseudo-random numbers
 l3tol(3C) — convert between 3-byte integers and long integers
 sputl(3X) — access long integer data in a machine-independent fashion
 strtol(3C) — convert strings to integer

loopback (software)

lo(5) — software loopback network interface

lost+found

mklost+found(1M) — makes a directory named lost+found to be used by fsck

Macintosh desktop

CommandShell(1) — manages command-interpretation windows and moderates access to the A/UX console window

Macintosh dialog boxes

Login(1M) — logs you in to A/UX by using a graphical user interface
 macquery(1M) — posts a Macintosh alert box to query the user

Macintosh environment, establishing preferences

changesize(1) — changes or displays the fields of the ‘SIZE’ resource of a file
 mactois(1) — convert between Macintosh encoding and International Standards Organization (ISO) encoding
 systemfolder(1) — create a personal System Folder
 keyset(1M) — sets the keyboard for the console

Macintosh or Macintosh-related applications

CommandShell(1) — manages command-interpretation windows and moderates access to the A/UX console window
 TextEditor(1) — lets you edit files interactively through mouse and menu operations
 changesize(1) — changes or displays the fields of the ‘SIZE’ resource of a file
 derez(1) — decompiles a resource file
 launch(1) — runs a Macintosh binary application in A/UX
 mactois(1) — convert between Macintosh encoding and International Standards Organization (ISO) encoding
 rez(1) — compiles Macintosh resource files from source code
 setfile(1) — sets attributes for Macintosh files, such as file type and creator

`systemfolder(1)` — create a personal System Folder
`Login(1M)` — logs you in to A/UX by using a graphical user interface
`keyset(1M)` — sets the keyboard for the console
`macquery(1M)` — posts a Macintosh alert box to query the user

Macintosh resources

`derez(1)` — decompiles a resource file
`fcnvt(1)` — converts a file in one storage format to a different storage format
`rez(1)` — compiles Macintosh resource files from source code
`setfile(1)` — sets attributes for Macintosh files, such as file type and creator

Macintosh toolbox

`slots(3X)` — provides ROM library functions

Macintosh user interface

`cmdo(1)` — builds command lines interactively
`macquery(1M)` — posts a Macintosh alert box to query the user

macros, format

`checkmm(1)` — check documents formatted with the `mm` macros
`m4(1)` — processes macros for C and other languages
`macref(1)` — produces a cross-reference listing of macro files
`mm(1)` — formats documents that contain `nroff` and `mm` macro formatting requests
`man(5)` — macros for formatting entries in this manual
`me(5)` — macros for formatting papers
`mm(5)` — macro package for formatting documents
`mptx(5)` — the macro package for formatting a permuted index
`ms(5)` — text formatting macros
`mv(5)` — a `troff` macro package for typesetting viewgraphs and slides

magic numbers

`a.out(4)` — common assembler and link editor output
`magic(4)` — magic number file for `file` command

magnetic tape

`mt(1)` — manipulates magnetic tape media
`tar(1)` — copies files to or from a `tar` archive
`tcb(1)` — blocks data to 8K for direct input to `/dev/rmt/tcx`
`tp(1)` — copies files to or from a `tp` archive
`tar(4)` — format of `tar` header
`mtio(7)` — provides an interface library for magnetic tape devices
`tc(7)` — tape device driver

mail handling

`biff(1)` — enables and disables notification of mail by `comsat`
`from(1)` — displays the mail header lines in your mailbox
`mail(1)` — send mail to users or read mail
`mailx(1)` — enables you to send and receive messages electronically

mesg(1) — permits or denies the receipt of messages
rmail(1) — handles remote mail received via UUCP
talk(1N) — talks to another user via the terminal
write(1) — writes to another user
comsat(1M) — invokes the server for biff
mailq(1M) — lists the contents of the mail queue
newaliases(1M) — rebuilds the database for the mail aliases file
sendmail(1M) — sends mail
aliases(4) — address and alias format used by sendmail

mail system, maintenance of

rmail(1) — handles remote mail received via UUCP
comsat(1M) — invokes the server for biff
mailq(1M) — lists the contents of the mail queue
newaliases(1M) — rebuilds the database for the mail aliases file
sendmail(1M) — sends mail

manual pages

apropos(1) — locates commands by keyword
man(1) — displays the named manual page entries
whatis(1) — reports a brief description for the manual page entry
specified
whereis(1) — reports the locations of the source, binary, and online help
files for a specified command
man(5) — macros for formatting entries in this manual

masks

sigblock(2) — block signals
sigpause(2) — release blocked signals and wait for interrupt
sigsetmask(2) — set current signal mask
umask(2) — set and get file creation mask
sigprocmask(3P) — examines and changes blocked signals

mastermind

mastermind(6) — plays the game of Mastermind

mathematical text

deroff(1) — removes nroff/troff, tbl, and eqn constructs
eqn(1) — format mathematical text for troff
neqn(1) — formats mathematical text for nroff
eqnchar(5) — special character definitions for eqn and neqn

mathematics

bc(1) — processes an arbitrary-precision arithmetic language
dc(1) — desk calculator
abs(3C) — return integer absolute value
abs(3F) — Fortran absolute value
acos(3F) — Fortran arccosine intrinsic function
aimag(3F) — Fortran imaginary part of complex argument
aint(3F) — Fortran integer part intrinsic function

asin(3F) — Fortran arcsine intrinsic function
 atan2(3F) — Fortran arctangent intrinsic function
 atan(3F) — Fortran arctangent intrinsic function
 atof(3C) — converts an ASCII string to floating-point number
 bessel(3M) — Bessel functions
 bool(3F) — Fortran bitwise boolean functions
 conjg(3F) — Fortran complex conjugate intrinsic function
 cos(3F) — Fortran cosine intrinsic function
 cosh(3F) — Fortran hyperbolic cosine intrinsic function
 dim(3F) — Fortran positive difference intrinsic functions
 dprod(3F) — Fortran double precision product intrinsic function
 ecvt(3C) — convert floating-point number to string
 exp(3F) — Fortran exponential intrinsic function
 exp(3M) — provide exponential, logarithm, power, and square root functions
 floor(3M) — floor, ceiling, remainder, absolute value functions
 frexp(3C) — manipulate parts of floating-point numbers
 gamma(3M) — logs a gamma function
 hypot(3M) — provides the Euclidean distance function
 l3tol(3C) — convert between 3-byte integers and long integers
 log10(3F) — Fortran common logarithm intrinsic function
 log(3F) — Fortran natural logarithm intrinsic function
 matherr(3M) — provides an error-handling function
 max(3F) — provides Fortran maximum-value functions
 min(3F) — provide Fortran minimum-value functions
 mod(3F) — provide Fortran remaindering intrinsic functions
 rand(3C) — call a simple random-number generator
 rand(3F) — provide a Fortran uniform random-number generator
 round(3F) — provide Fortran nearest integer functions
 sign(3F) — returns Fortran transfer-of-sign intrinsic functions
 sin(3F) — provide Fortran sine intrinsic functions
 sinh(3F) — provide Fortran hyperbolic sine intrinsic function
 sinh(3M) — provide hyperbolic functions
 sputl(3X) — access long integer data in a machine-independent fashion
 sqrt(3F) — provide Fortran square root intrinsic functions
 strtod(3C) — converts a string to a double-precision number
 strtol(3C) — convert strings to integer
 tan(3F) — Fortran tangent intrinsic function
 tanh(3F) — Fortran hyperbolic tangent intrinsic function
 trig(3M) — provide trigonometric functions
 math(5) — math functions and constants

maximum values

`max(3F)` — provides Fortran maximum-value functions

maze

`maze(6)` — generates a maze

memory, general

`pagesize(1)` — displays the system page size

`swap(1M)` — adds disk blocks to or deletes them from the swap area

`brk(2)` — change data segment space allocation

`phys(2)` — allows a process to access physical addresses

`plock(2)` — enables a lock process for text or data in memory

`end(3C)` — last locations in program

`malloc(3C)` — provide a main memory allocator

`malloc(3X)` — provide a fast main memory allocator

`memory(3C)` — perform memory operations

`core(4)` — format of core image file

`mem(7)` — provide an interface for access to core memory

memory, shared

`mkshlib(1)` — creates a shared library

`shmctl(2)` — shared memory control operations

`shmget(2)` — get shared memory segment

`shmop(2)` — shared memory operations

merging files

`cat(1)` — catenates and displays the contents of files

`join(1)` — combines (joins) two relational files

`merge(1)` — merges three files into one

`paste(1)` — merges lines of several files or subsequent lines of one file

`soelim(1)` — eliminates the source commands from `nroff` input

`sort(1)` — sorts or merges files

`tsort(1)` — sorts lines in a file topologically

`acctmerg(1M)` — merges or adds accounting files

message queue

`msgget(2)` — gets message queue

messages

`ipcrm(1)` — removes interprocess communications facilities

`ipcs(1)` — reports interprocess communication facilities status

`msg(1)` — permits or denies the receipt of messages

`write(1)` — writes to another user

`msgctl(2)` — message control operations

`msgget(2)` — gets message queue

`msgop(2)` — message operations

`recv(2N)` — receive a message from a socket

`send(2N)` — send a message from a socket

minimum values

`min(3F)` — provide Fortran minimum-value functions

modems

`ct(1C)` — runs `login` on a dial-up line
`cu(1C)` — establishes an interactive connection with another system
`kermit(1C)` — invokes the Kermit file-transfer program
`tip(1C)` — establishes a connection to a remote system
`uucp(1C)` — copies files from one system to another system
`uux(1C)` — runs a command on a remote system
`slip(1M)` — assigns a serial line to a network interface
`uucico(1M)` — transfers files as specified by `uucp` work files
`dial(3C)` — establishes an out-going terminal line connection
`dialup(4)` — modem escape sequence file
`phones(4)` — remote host telephone number database

modification times, file

`ls(1)` — lists the contents of a directory
`touch(1)` — updates access and modification times of a file
`utime(2)` — set file access and modification times

monitor processing

`300(1)` — filter text containing printer control sequences for a DASI terminal
`4014(1)` — filters text containing printer control sequences a page at a time
`450(1)` — filters text containing printer control sequences for the DASI terminal
`col(1)` — filters text containing printer control sequences for use at a display device
`colcrt(1)` — filters `nroff` output for terminal previewing
`greek(1)` — filters text for vintage display devices
`tc(1)` — interprets `troff` output for use at a vintage display device
`tplot(1G)` — interprets plotter instructions for use at a vintage display device
`ul(1)` — filters special underlining sequences imbedded in text for use at a display device

moo

`moo(6)` — plays the game of moo

Motorola S-records

`hex(1)` — converts an object file to Motorola S-record format
`rcvhex(1)` — receives and converts Motorola S-records from a port to a file

mounting file systems

`automount(1M)` — mounts Network File System (NFS) when needed
`mount(1M)` — mount and unmount file systems
`mountd(1M)` — invokes the Network File System (NFS) mount-request

server
 showmount(1M) — shows all remote mounts
 umount(2) — unmount a file system
 mount(3) — mounts a file system
 mount(3N) — keeps track of remotely mounted file systems
 fstab(4) — parameter file format
 mtab(4) — mounted file system table

mouse
 mouse(7) — provides a mouse input device driver

moving files
 mv(1) — moves or renames files

multiplexing
 select(2N) — synchronous I/O multiplexing

multiplication
 dprod(3F) — Fortran double precision product intrinsic function

name binding
 yperv(1M) — provide Network Information Service (NIS) service
 bind(2N) — bind a name to a socket
 HOSTNAME(4) — host name and domain name database

Name Binding Protocol
 nbp(3N) — perform AppleTalk Name Binding Protocol (NBP) interface operations

name cache
 ncstats(1M) — displays kernel name cache statistics

Name Information Server
 domainname(1) — sets or displays the name of the Network Information Service (NIS) domain
 ypcat(1) — lists the contents of a Network Information Service (NIS) map
 ypmatch(1) — lists the value of a specified key in a Network Information Service (NIS) map
 yppasswd(1) — changes a login password on the Network Information Service (NIS) master server
 ypwhich(1) — displays the host name of a system's Network Information Service (NIS) server
 makedbm(1M) — generates a Network Information Service (NIS) dbm file
 revnetgroup(1M) — reverses the netgroup file
 ypinit(1M) — initializes Network Information Service (NIS) maps for master and slave servers
 ypmake(1M) — rebuilds the Network Information Service (NIS) maps
 yppasswdd(1M) — handle requests to change a password served by the Network Information Service (NIS)
 yppoll(1M) — reports the version of a Network Information Service

(NIS) map that is on an NIS server
yppush(1M) — propagates changed Network Information Service (NIS) maps
ypserv(1M) — provide Network Information Service (NIS) service
ypset(1M) — sets ypbind to a particular domain and Network Information Service (NIS) server
ypxfr(1M) — transfers a Network Information Service (NIS) map to the local system
ypclnt(3N) — provide a Network Information Service (NIS) client interface
yppasswd(3N) — updates a user password on the Network Information Service (NIS) master server
ethers(4) — Ethernet address to host name database or YP domain
ypfiles(4) — the Network Information Service (NIS) database and directory structure

Name Information Server maps

ypwhich(1) — displays the host name of a system's Network Information Service (NIS) server
ypoll(1M) — reports the version of a Network Information Service (NIS) map that is on an NIS server
yppush(1M) — propagates changed Network Information Service (NIS) maps
ypxfr(1M) — transfers a Network Information Service (NIS) map to the local system

name servers

nslookup(1) — interactively queries name servers
ypcat(1) — lists the contents of a Network Information Service (NIS) map

NBP

nbp(3N) — perform AppleTalk Name Binding Protocol (NBP) interface operations

network bridges

rtmp(3N) — identify AppleTalk node and bridge addresses

network domains

ypwhich(1) — displays the host name of a system's Network Information Service (NIS) server
ypset(1M) — sets ypbind to a particular domain and Network Information Service (NIS) server
getdomainname(2N) — get/set name of current network domain

Network File System

domainname(1) — sets or displays the name of the Network Information Service (NIS) domain
automount(1M) — mounts Network File System (NFS) when needed
exportfs(1M) — exports and unexports directories to Network File

System (NFS) clients

lockd(1M) — handle local and remote lock requests
mountd(1M) — invokes the Network File System (NFS) mount-request server
nfsd(1M) — invoke the NFS daemons
nfsstat(1M) — displays Network File System (NFS) statistics
rpcinfo(1M) — reports RPC information
showmount(1M) — shows all remote mounts
spray(1M) — sprays packets
sprayd(1M) — returns information for the `spray` command
statd(1M) — provide crash and recovery monitoring for network locking services
fsmount(2) — mount a network file system (NFS)
nfssvc(2) — provides NFS daemons
exportent(3) — get exported file-system information
exports(4) — directories to export to Network File System (NFS) clients
fstab(4) — parameter file format

network groups

ypcat(1) — lists the contents of a Network Information Service (NIS) map
revnetgroup(1M) — reverses the `netgroup` file
getnetgrent(3N) — get network group entry
netgroup(4) — list of network groups

network maintenance, Name Information Server

ypcat(1) — lists the contents of a Network Information Service (NIS) map
ypmatch(1) — lists the value of a specified key in a Network Information Service (NIS) map
yppasswd(1) — changes a login password on the Network Information Service (NIS) master server
ypwhich(1) — displays the host name of a system's Network Information Service (NIS) server
makedbm(1M) — generates a Network Information Service (NIS) dbm file
revnetgroup(1M) — reverses the `netgroup` file
ypinit(1M) — initializes Network Information Service (NIS) maps for master and slave servers
ypmake(1M) — rebuilds the Network Information Service (NIS) maps
yppasswdd(1M) — handle requests to change a password served by the Network Information Service (NIS)
yppoll(1M) — reports the version of a Network Information Service (NIS) map that is on an NIS server
yppush(1M) — propagates changed Network Information Service (NIS) maps

ypserv(1M) — provide Network Information Service (NIS) service
ypset(1M) — sets ypbind to a particular domain and Network
Information Service (NIS) server
ypxfr(1M) — transfers a Network Information Service (NIS) map to the
local system

network maintenance, UUCP system

Uutry(1M) — contacts a remote system with debugging on
uucheck(1M) — checks the uucp directories and files
uucico(1M) — transfers files as specified by uucp work files
uucleanup(1M) — removes old files from the uucp spool directory
uucpd(1M) — handles the transfer of files by uucico over TCP/IP
connections
uudemon.admin(1M) — mails current uucp work status to the uucp
administrator
uudemon.cleanup(1M) — cleans up files in the uucp spool directory
uudemon.hour(1M) — processes spooled uucp requests
uudemon.poll(1M) — sets up polling for selected systems
uusched(1M) — schedules uucp file transfers

network protocols

getprotoent(3N) — get a protocol entry
protocols(4N) — protocol name database

network, status

rup(1N) — displays the status of machines on the local network (RPC
version)
ruptime(1N) — displays the host status of local machines
rusers(1N) — produces a login list for local machines (RPC version)
rwho(1N) — displays a list of the active users from all of the systems on
the local network
ether(3N) — monitors Ethernet traffic
sm_inter(3N) — status monitor protocol

network testing

ping(1M) — exercises the TCP/IP network by sending Internet Control
Message Protocol (ICMP) packets to a named host
lo(5) — software loopback network interface

networks, general

atstatus(1) — displays status information from an AppleTalk device
checkinstall(1) — checks the installation of boards
netstat(1N) — displays network status information
ypcat(1) — lists the contents of a Network Information Service (NIS)
map
ypmatch(1) — lists the value of a specified key in a Network Information
Service (NIS) map
yppasswd(1) — changes a login password on the Network Information
Service (NIS) master server

ypwhich(1) — displays the host name of a system's Network Information Service (NIS) server
 appleping(1M) — exercises the AppleTalk network by sending packets to a named host
 appletalk(1M) — enables you to configure and display AppleTalk network interfaces
 ifconfig(1M) — manages network interfaces
 lockd(1M) — handle local and remote lock requests
 ping(1M) — exercises the TCP/IP network by sending Internet Control Message Protocol (ICMP) packets to a named host
 route(1M) — manipulates the routing tables
 routed(1M) — invokes the network routing daemon
 rwall(1M) — writes to all users over a network
 rwalld(1M) — invokes the network rwall server
 slattach(1M) — attaches a serial line to a network interface
 slattconf(1M) — attaches a serial line to a network interface and configures the network interface
 slip(1M) — assigns a serial line to a network interface
 statd(1M) — provide crash and recovery monitoring for network locking services
 ypinit(1M) — initializes Network Information Service (NIS) maps for master and slave servers
 ypmake(1M) — rebuilds the Network Information Service (NIS) maps
 yppasswdd(1M) — handle requests to change a password served by the Network Information Service (NIS)
 yppoll(1M) — reports the version of a Network Information Service (NIS) map that is on an NIS server
 yppush(1M) — propagates changed Network Information Service (NIS) maps
 ypserv(1M) — provide Network Information Service (NIS) service
 ypset(1M) — sets ypbind to a particular domain and Network Information Service (NIS) server
 ypxfr(1M) — transfers a Network Information Service (NIS) map to the local system
 connect(2N) — initiates a connection on a socket
 socket(2N) — create an endpoint for communication
 socketpair(2) — creates a pair of connected sockets
 byteorder(3N) — convert values between host and network byte order
 gethostbyaddr(3N) — get network host entry
 getnetent(3N) — get network entry
 rwall(3N) — writes to specified remote machines
 ypclnt(3N) — provide a Network Information Service (NIS) client interface
 yppasswd(3N) — updates a user password on the Network Information

Service (NIS) master server

NETADDRS(4) — network address database

appletalkrc(4) — obsolete AppleTalk network configuration file

ypfiles(4) — the Network Information Service (NIS) database and directory structure

lo(5) — software loopback network interface

news

news(1) — displays local news items

NFS

automount(1M) — mounts Network File System (NFS) when needed

exportfs(1M) — exports and unexports directories to Network File System (NFS) clients

mountd(1M) — invokes the Network File System (NFS) mount-request server

nfsd(1M) — invoke the NFS daemons

nfsstat(1M) — displays Network File System (NFS) statistics

fsmount(2) — mount a network file system (NFS)

nfssvc(2) — provides NFS daemons

exportent(3) — get exported file-system information

exports(4) — directories to export to Network File System (NFS) clients

fstab(4) — parameter file format

nodes

uname(1) — displays identification information about the current system

chgnod(1M) — changes the current A/UX system node name

mknod(1M) — builds a device file

rtmp(3N) — identify AppleTalk node and bridge addresses

intro(7) — introduces device drivers and interfaces

notification (mail)

biff(1) — enables and disables notification of mail by comsat

nroff

checknr(1) — checks nroff/troff files

colcrt(1) — filters nroff output for terminal previewing

deroff(1) — removes nroff/troff, tbl, and eqn constructs

diffmk(1) — marks the differences between two files

mm(1) — formats documents that contain nroff and mm macro formatting requests

neqn(1) — formats mathematical text for nroff

nroff(1) — text formatter

soelim(1) — eliminates the source commands from nroff input

tbl(1) — table formatter for nroff or troff

eqnchar(5) — special character definitions for eqn and neqn

mptx(5) — the macro package for formatting a permuted index

ms(5) — text formatting macros

nterm(5) — terminal driving tables for nroff

null device

`null(7)` — represents the null device file

numbers

`arithmetic(6)` — provides arithmetic problems

`number(6)` — converts Arabic numerals to English

numeric sign

`sign(3F)` — returns Fortran transfer-of-sign intrinsic functions

NVE

`atlookup(1)` — looks up network-visible entities (NVEs) registered on the AppleTalk network system

object file

`conv(1)` — swaps bytes in COFF files

`dump(1)` — stores (saves) selected parts of an object file

`ld(1)` — invokes the link editor for common object files

`nm(1)` — displays the symbol table of a common object file

`strings(1)` — finds the printable strings in an object or other binary file

`cpset(1M)` — installs files in specified directories

`ldclose(3X)` — close a common object file

`ldfcn(3X)` — provide common object file access routines

`ldfhread(3X)` — read the file header of a common object file

`ldgetname(3X)` — retrieves symbol name for object file symbol table entry

`ldlread(3X)` — manipulate line number entries of a common object file function

`ldlseek(3X)` — seek to line number entries of a section of a common object file

`ldohseek(3X)` — seek to the optional file header of a common object file

`ldopen(3X)` — open a common object file for reading

`ldrseek(3X)` — seek to relocation entries of a section of a common object file

`ldshread(3X)` — read an indexed/named section header of a common object file

`ldsseek(3X)` — seek to an indexed/named section of a common object file

`ldtbindex(3X)` — compute index of a symbol table entry of a common object file

`ldtbread(3X)` — read an indexed symbol table entry of a common object file

`ldtbseek(3X)` — seek to the symbol table of a common object file

`nlist(3C)` — gets entries from name list

`a.out(4)` — common assembler and link editor output

`aouthdr(4)` — `a.out` header for common object files

`filehdr(4)` — file header for common object files

`linenum(4)` — line number entries in a common object file

reloc(4) — relocation information for a common object file

scnhdr(4) — section header for a common object file

syms(4) — common object file symbol table format

octal

od(1) — converts binary data to a displayable form in octal, decimal, hexadecimal, or ASCII

online documentation

apropos(1) — locates commands by keyword

man(1) — displays the named manual page entries

whatis(1) — reports a brief description for the manual page entry specified

whereis(1) — reports the locations of the source, binary, and online help files for a specified command

man(5) — macros for formatting entries in this manual

optimization

cc(1) — invokes the C compiler

prof(1) — displays profile data

dcopy(1M) — copies System V File System-style file systems for optimal access time

kconfig(1M) — tunes kernel parameters for work-load optimization

sadc(1M) — report system activity

tunefs(1M) — tunes a Berkeley 4.2 (UFS) file system

profil(2) — reports the execution time of an application

courses5.0(3X) — provides BSD-style screen functions with optimal cursor motion

courses(3X) — CRT screen handling and optimization package

overviews

intro(1) — introduces the command and application programs

rcsintro(1) — introduces RCS commands

acct(1M) — present an overview of accounting commands

intro(1M) — introduces system maintenance commands

intro(2) — introduces system calls and error numbers

intro(3) — introduces the subroutines and libraries

intro(4) — introduction to file formats

intro(5) — introduction to miscellaneous facilities

intro(6) — introduction to games

intro(7) — introduces device drivers and interfaces

intro(8) — introduces commands executed from the A/UX Startup shell

ownership, file

chown(1) — change the owner or group of a file

ls(1) — lists the contents of a directory

chown(2) — changes the owner and group of a file

packets

- spray(1M) — sprays packets
- sprayd(1M) — returns information for the spray command
- spray(3N) — scatters data in order to check the network

pagination

- 4014(1) — filters text containing printer control sequences a page at a time
- daps(1) — invokes the Autologic APS-5 phototypesetter troff post-processor
- enscript(1) — converts text files to format for printing
- mm(1) — formats documents that contain nroff and mm macro formatting requests
- mmt(1) — typeset documents that contain troff and mm or mv macro-formatting requests
- mvt(1) — typeset documents that contain troff and mm or mv macro-formatting requests
- nroff(1) — text formatter
- otroff(1) — formats text for a specific phototypesetter
- pr(1) — formats text for a print device
- psdit(1) — converts troff intermediate format to POSTSCRIPT format
- psroff(1) — formats a file through troff so it can be printed on a POSTSCRIPT printer
- roffbib(1) — prints out all records in a bibliographic database
- troff(1) — formats and typesets files

PAP

- atprint(1) — transfers data to a printer by using AppleTalk protocols
- atstatus(1) — displays status information from an AppleTalk device
- pap(3N) — provide AppleTalk Printer Access Protocol (PAP) interface

parser

- awk(1) — scans a file for lines that match a specific pattern
- getopt(1) — parses command options
- lex(1) — generates programs for simple lexical tasks
- yacc(1) — compiles compilers (yet another compiler-compiler)

partitions

- dd(1) — converts and copies a file
- dp(1M) — performs disk partitioning
- pname(1M) — associates named partitions with device files
- getptabent(3) — get partition table file entry
- bzb(4) — Block Zero Block file format
- dpme(4) — format of disk partition map entries
- ptab(4) — partition table file

password file

`finger(1)` — displays information about the users on a system
`pwck(1M)` — check the password/group files
`vipw(1M)` — edits the password file
`yppasswdd(1M)` — handle requests to change a password served by the
Network Information Service (NIS)
`getpwent(3C)` — get the password file entry
`putpwent(3C)` — write password file entry
`passwd(4)` — password file

passwords

`crypt(1)` — encodes and decodes passwords
`passwd(1)` — changes the login password
`yppasswd(1)` — changes a login password on the Network Information
Service (NIS) master server
`getpass(3C)` — read a password
`getpwent(3C)` — get the password file entry
`putpwent(3C)` — write password file entry
`yppasswd(3N)` — updates a user password on the Network Information
Service (NIS) master server
`auxstartuprc(4)` — authorization file that helps password-protect and
otherwise secure A/UX Startup

path string functions

`basename(1)` — get part of a pathname
`realpath(3)` — returns the real filename of a file

pathnames

`basename(1)` — get part of a pathname
`whereis(1)` — reports the locations of the source, binary, and online help
files for a specified command
`pathconf(3P)` — get configurable pathname variables
`realpath(3)` — returns the real filename of a file

patterns

`awk(1)` — scans a file for lines that match a specific pattern
`grep(1)` — search a file for a specific pattern
`regexp(5)` — regular expression compile and match routines

pause

`shl(1)` — manages the layering of multiple shells
`sleep(1)` — suspends the system for a specified interval of time
`sigpause(2)` — release blocked signals and wait for interrupt
`wait3(2N)` — wait for child process to stop or terminate
`wait(2)` — wait for child process to stop or terminate
`sigsuspend(3P)` — waits for a signal
`sleep(3C)` — suspends execution for interval
`tcdrain(3P)` — provide line control functions
`usleep(3)` — suspend execution for interval

PDP-11 computer

swab(3C) — swaps bytes

peer

getpeername(2N) — gets the name of a connected peer

performance

cc(1) — invokes the C compiler

nice(1) — executes a command at low priority

prof(1) — displays profile data

timex(1) — reports the elapsed, user, and system time during the execution of a command

kconfig(1M) — tunes kernel parameters for work-load optimization

profil(2) — reports the execution time of an application

monitor(3C) — prepares an execution profile

peripheral device files

tty(1) — obtains the device filename for the terminal or CommandShell window where it is invoked

dev_kill(1M) — removes device files from a directory

devnm(1M) — displays the current device name

mknod(1M) — builds a device file

pname(1M) — associates named partitions with device files

tty(7) — controls the terminal interface

permissions

chmod(1) — changes the permissions of a file

chown(1) — change the owner or group of a file

chmod(2) — change mode of file

umask(2) — set and get file creation mask

permuted index

mptx(5) — the macro package for formatting a permuted index

pi

math(5) — math functions and constants

pipe

tee(1) — transcribes data

pipe(2) — creates an interprocess channel

popen(3S) — initiate pipe to/from a process

plotters

pac(1M) — gathers printer/plotter accounting information

plotting

graph(1G) — draws a graph

spline(1G) — interpolates a smooth curve

tplot(1G) — interprets plotter instructions for use at a vintage display device

plot(3X) — provide graphics interface subroutines

plot(4) — graphics interface

portability

- `ar(1)` — maintains a library of files in an archive
- `lint(1)` — invokes a C program checker
- `pax(1)` — copies files to or from an archive in an IEEE format

ports

- `ct(1C)` — runs `login` on a dial-up line
- `cu(1C)` — establishes an interactive connection with another system
- `kermit(1C)` — invokes the Kermit file-transfer program
- `stty(1)` — sets the modes for a terminal
- `tip(1C)` — establishes a connection to a remote system
- `tty(1)` — obtains the device filename for the terminal or CommandShell window where it is invoked
- `updater(1)` — updates files between two machines
- `getty(1M)` — set the initial communication modes, such as speed and line discipline, for the purpose of logging users in to A/UX through serial lines
- `setport(1M)` — sets the characteristics of a serial port
- `slattach(1M)` — attaches a serial line to a network interface
- `slattconf(1M)` — attaches a serial line to a network interface and configures the network interface
- `slip(1M)` — assigns a serial line to a network interface
- `gettydefs(4)` — speed and terminal settings used by `getty`
- `inittab(4)` — script for the `init` process
- `ttytype(4)` — database of terminal types by port
- `serial(7)` — provides the on-board serial ports

POSIX compatibility

- `setposix(3P)` — sets POSIX compatibility flags

poster-size text

- `banner7(1)` — generates a large banner
- `banner(1)` — generates a poster

posters, printing text for

- `banner7(1)` — generates a large banner
- `banner(1)` — generates a poster

PostScript®

- `enscript(1)` — converts text files to format for printing
- `psdit(1)` — converts `troff` intermediate format to POSTSCRIPT format
- `psroff(1)` — formats a file through `troff` so it can be printed on a POSTSCRIPT printer
- `transcript(1M)` — filter data for the POSTSCRIPT printers
- `afm(4)` — Adobe POSTSCRIPT font metrics file format
- `postscript(4)` — POSTSCRIPT print file format

power

`powerdown(1M)` — turns off power to the computer

preferences, Macintosh

`changesize(1)` — changes or displays the fields of the 'SIZE' resource of a file

`mactois(1)` — convert between Macintosh encoding and International Standards Organization (ISO) encoding

`systemfolder(1)` — create a personal System Folder

`keyset(1M)` — sets the keyboard for the console

preprocessors, text

`awk(1)` — scans a file for lines that match a specific pattern

`col(1)` — filters text containing printer control sequences for use at a display device

`comm(1)` — selects or rejects lines common to two sorted files

`cpre(1)` — invokes the C language preprocessor

`cw(1)` — prepare constant-width text for `troff`

`daps(1)` — invokes the Autologic APS-5 phototypesetter `troff` post-processor

`deroff(1)` — removes `nroff/troff`, `tbl`, and `eqn` constructs

`eqn(1)` — format mathematical text for `troff`

`expand(1)` — expand tabs to spaces, and vice versa

`fmt(1)` — invokes a simple text formatter

`fold(1)` — folds long lines for finite-width output device

`grap(1)` — invokes a `pic` preprocessor for drawing graphs

`iw2(1)` — prepares data to be printed on the Apple ImageWriter II printer

`m4(1)` — processes macros for C and other languages

`neqn(1)` — formats mathematical text for `nroff`

`pic(1)` — preprocesses `troff` files that contain drawings

`pr(1)` — formats text for a print device

`rev(1)` — reverses characters within each line of text

`soelim(1)` — eliminates the source commands from `nroff` input

`sort(1)` — sorts or merges files

`ssp(1)` — produces single spaced output

`tabs(1)` — sets the tab stops on a terminal

`tbl(1)` — table formatter for `nroff` or `troff`

`uniq(1)` — reports repeated lines in a file

pretty printing

`cb(1)` — improves spacing and indentation of C source files

`indent(1)` — indents and formats C program source

Print Access Protocol

`atprint(1)` — transfers data to a printer by using AppleTalk protocols

`atstatus(1)` — displays status information from an AppleTalk device

`pap(3N)` — provide AppleTalk Printer Access Protocol (PAP) interface

print spooler maintenance

`enable(1)` — enable or disable LP printers
`lpstat(1)` — prints lp status information
`accept(1M)` — allows lp requests
`lpadmin(1M)` — configures the lp spooling system
`lpc(1M)` — controls the operation of the line printer
`lpd(1M)` — supports the Berkeley print spooler ." 4.2 line-printer daemon
`lpsched(1M)` — start or stop the lp request scheduler and move requests
`lpctest(1M)` — generates a line-printer ripple pattern
`reject(1M)` — prevents LP requests
`transcript(1M)` — filter data for the POSTSCRIPT printers

printer testing

`lpctest(1M)` — generates a line-printer ripple pattern

printers, general

`asa(1)` — interprets ASA carriage control characters
`at_cho_prn(1)` — allows you to choose a default printer on the
 AppleTalk internet
`cancel(1)` — cancels print requests spooled through the lp command
`enable(1)` — enable or disable LP printers
`lp(1)` — spools print requests to printers
`lpq(1)` — queries the print spooler for progress information
`lpr(1)` — spools print requests to printers
`lprm(1)` — removes jobs from the line printer spooling queue for a
 Berkeley file system (4.2)
`lpstat(1)` — prints lp status information
`accept(1M)` — allows lp requests
`lpadmin(1M)` — configures the lp spooling system
`lpc(1M)` — controls the operation of the line printer
`lpd(1M)` — supports the Berkeley print spooler ." 4.2 line-printer daemon
`lpsched(1M)` — start or stop the lp request scheduler and move requests
`lpctest(1M)` — generates a line-printer ripple pattern
`pac(1M)` — gathers printer/plotter accounting information
`reject(1M)` — prevents LP requests

printing, Appletalk

`at_cho_prn(1)` — allows you to choose a default printer on the
 AppleTalk internet
`atlookup(1)` — looks up network-visible entities (NVEs) registered on
 the AppleTalk network system
`atprint(1)` — transfers data to a printer by using AppleTalk protocols
`atstatus(1)` — displays status information from an AppleTalk device

printing files

`cancel(1)` — cancels print requests spooled through the lp command
`lp(1)` — spools print requests to printers
`lpq(1)` — queries the print spooler for progress information

lpr(1) — spools print requests to printers
 lprm(1) — removes jobs from the line printer spooling queue for a Berkeley file system (4.2)

printing, poster-size text

banner7(1) — generates a large banner
 banner(1) — generates a poster

priority (process)

nice(1) — executes a command at low priority
 nice(2) — changes the priority of a process

process accounting

lav(1) — displays load average statistics
 acctcms(1M) — summarizes commands from per-process accounting records
 acctcom(1M) — searches and formats process accounting files
 acctprc(1M) — provide process accounting
 acct(2) — enable or disable process accounting
 times(2) — get process and child process times
 acct(4) — per-process accounting file format
 prof(5) — profile within a function

process groups

getpid(2) — get process, process group, or parent process IDs
 killpg(3N) — sends signal to a process group
 tcgetpgrp(3P) — gets distinguished process group ID
 tcsetpgrp(3P) — sets distinguished process group ID

process IDs

ps(1) — reports process status
 getpid(2) — get process, process group, or parent process IDs

process limits

kconfig(1M) — tunes kernel parameters for work-load optimization
 ulimit(2) — get and set user limits

process priority

nice(1) — executes a command at low priority
 nice(2) — changes the priority of a process

process scheduling

at(1) — run commands at a later time
 crontab(1) — aids in the use of the cron process scheduling program
 nice(1) — executes a command at low priority
 cron(1M) — runs the clock daemon
 alarm(2) — sets a process's alarm clock

process termination

kill(1) — terminates a process
 nohup(1) — runs a command so that it can continue to run even after your session has ended
 killall(1M) — kills all active processes

shutdown(1M) — terminates processes that support multi-user mode and enters single-user mode

exit(2) — terminate process

abort(3C) — generates an IOT fault

abort(3F) — terminates a Fortran program

processes, general

kill(1) — terminates a process

ps(1) — reports process status

fuser(1M) — identifies processes using a file or file structure

init(1M) — spawn general processes

killall(1M) — kills all active processes

lockd(1M) — handle local and remote lock requests

exit(2) — terminate process

fork(2) — creates a new process

getpid(2) — get process, process group, or parent process IDs

kill(2) — sends a signal to a process or a group of processes

nice(2) — changes the priority of a process

pause(2) — suspends a process until signal

phys(2) — allows a process to access physical addresses

pipe(2) — creates an interprocess channel

plock(2) — enables a lock process for text or data in memory

ptrace(2) — process trace

setcompat(2) — set or get process compatibility mode

wait3(2N) — wait for child process to stop or terminate

wait(2) — wait for child process to stop or terminate

killpg(3N) — sends signal to a process group

popen(3S) — initiate pipe to/from a process

processes, monitoring

time(1) — prints the elapsed time during the execution of a command

timex(1) — reports the elapsed, user, and system time during the execution of a command

processes, signaling

ipcrm(1) — removes interprocess communications facilities

kill(1) — terminates a process

processing unit

machid(1) — provide truth values about processor type

values(5) — machine-dependent values

processors, text

awk(1) — scans a file for lines that match a specific pattern

col(1) — filters text containing printer control sequences for use at a display device

comm(1) — selects or rejects lines common to two sorted files

cpp(1) — invokes the C language preprocessor

daps(1) — invokes the Autologic APS-5 phototypesetter troff post-

processor

`deroff(1)` — removes `nroff/troff`, `tbl`, and `eqn` constructs
`eqn(1)` — format mathematical text for `troff`
`expand(1)` — expand tabs to spaces, and vice versa
`fmt(1)` — invokes a simple text formatter
`fold(1)` — folds long lines for finite-width output device
`grap(1)` — invokes a `pic` preprocessor for drawing graphs
`iw2(1)` — prepares data to be printed on the Apple ImageWriter II printer
`m4(1)` — processes macros for C and other languages
`neqn(1)` — formats mathematical text for `nroff`
`pic(1)` — preprocesses `troff` files that contain drawings
`pr(1)` — formats text for a print device
`rev(1)` — reverses characters within each line of text
`sort(1)` — sorts or merges files
`ssp(1)` — produces single spaced output
`tabs(1)` — sets the tab stops on a terminal
`tbl(1)` — table formatter for `nroff` or `troff`
`uniq(1)` — reports repeated lines in a file

program debugging

`adb(1)` — debugs executable programs
`ctrace(1)` — debugs a C program
`dbx(1)` — debugs and executes programs
`sdb(1)` — symbolic debugger

program source

`admin(1)` — creates and administers SCCS files
`cb(1)` — improves spacing and indentation of C source files
`cdc(1)` — changes the delta commentary of an SCCS delta
`ci(1)` — checks in RCS revisions
`co(1)` — checks out RCS revisions
`comb(1)` — combines SCCS deltas
`get(1)` — gets a version of an SCCS file
`help(1)` — provides help information about SCCS commands and messages
`ident(1)` — displays RCS keywords and their values
`indent(1)` — indents and formats C program source
`lint(1)` — invokes a C program checker
`make(1)` — maintains, updates, and regenerates groups of files
`prs(1)` — displays information about an SCCS file
`rccs(1)` — creates new RCS files or changes attributes of existing RCS files
`rccsdiff(1)` — compares RCS revisions
`rccsintro(1)` — introduces RCS commands
`rccsmerge(1)` — merges two versions of an RCS file
`rlog(1)` — displays log messages and other information about RCS files
`rmdel(1)` — removes a delta from an SCCS file

sact(1) — displays who has checked a Source Code Control System (SCCS) file out for editing
sccs(1) — performs SCCS subsystem commands
sccsdiff(1) — compares two versions of an SCCS file
ucbdiff3(1) — reports the differences between three files
ucbdiff(1) — reports differences between two files or directories
unget(1) — undoes a previous get of an SCCS file
val(1) — validate SCCS file
what(1) — reports identification information for a file
sccstorcs(1M) — builds an RCS file from an SCCS file
rscsfile(4) — format of an RCS file
sccsfile(4) — format of an SCCS file
programming, general development tools
adb(1) — debugs executable programs
admin(1) — creates and administers SCCS files
ar(1) — maintains a library of files in an archive
as(1) — assembles files by translating assembler mnemonics to object code
bs(1) — compiles and interprets `bs` programs
cdc(1) — changes the delta commentary of an SCCS delta
ci(1) — checks in RCS revisions
co(1) — checks out RCS revisions
comb(1) — combines SCCS deltas
conv(1) — swaps bytes in COFF files
dbx(1) — debugs and executes programs
delta(1) — makes a delta (change) to an SCCS file
dis(1) — produces an assembly language listing for a specified file
dump(1) — stores (saves) selected parts of an object file
get(1) — gets a version of an SCCS file
help(1) — provides help information about SCCS commands and messages
hex(1) — converts an object file to Motorola S-record format
ld(1) — invokes the link editor for common object files
lex(1) — generates programs for simple lexical tasks
lorder(1) — finds the ordering relation for an object library
make(1) — maintains, updates, and regenerates groups of files
mkshlib(1) — creates a shared library
nm(1) — displays the symbol table of a common object file
od(1) — converts binary data to a displayable form in octal, decimal, hexadecimal, or ASCII
prof(1) — displays profile data
prs(1) — displays information about an SCCS file
rscs(1) — creates new RCS files or changes attributes of existing RCS files
rscsdiff(1) — compares RCS revisions

rcsintro(1) — introduces RCS commands
 rcsmerge(1) — merges two versions of an RCS file
 rcvhex(1) — receives and converts Motorola S-records from a port to a file
 regcmp(1) — compiles regular expressions with a file
 rlog(1) — displays log messages and other information about RCS files
 rmdel(1) — removes a delta from an SCCS file
 rpcgen(1) — generates C source code from a remote procedure call (RPC) source file
 sact(1) — displays who has checked a Source Code Control System (SCCS) file out for editing
 sccs(1) — performs SCCS subsystem commands
 sccsdiff(1) — compares two versions of an SCCS file
 sdb(1) — symbolic debugger
 size(1) — displays section sizes of common object files
 strings(1) — finds the printable strings in an object or other binary file
 strip(1) — strips symbol and line number information from an object file
 tsort(1) — sorts lines in a file topologically
 unget(1) — undoes a previous get of an SCCS file
 val(1) — validate SCCS file
 vc(1) — manipulates version control information inside a data stream
 what(1) — reports identification information for a file
 yacc(1) — compiles compilers (yet another compiler-compiler)
 sccstorcs(1M) — builds an RCS file from an SCCS file

programming, Macintosh development tools

derez(1) — decompiles a resource file
 rez(1) — compiles Macintosh resource files from source code

programming, shell

basename(1) — get part of a pathname
 echo(1) — echoes its arguments
 expr(1) — evaluates arguments as an expression
 getopt(1) — parses command options
 line(1) — reads one line from the standard input
 query(1) — queries the user for input
 rev(1) — reverses characters within each line of text
 test(1) — evaluates conditions
 tput(1) — queries terminfo database
 true(1) — provides truth values
 macquery(1M) — posts a Macintosh alert box to query the user

programming, using C

cb(1) — improves spacing and indentation of C source files
 cc(1) — invokes the C compiler
 cflow(1) — generates a C flowgraph
 cpp(1) — invokes the C language preprocessor

ctags(1) — maintains a tags file for a C program
 ctrace(1) — debugs a C program
 cxref(1) — generates a C program cross-reference
 ident(1) — displays RCS keywords and their values
 indent(1) — indents and formats C program source
 lint(1) — invokes a C program checker
 mkstr(1) — creates an error message file by massaging C source
 programs
 xstr(1) — reports strings from C programs to implement shared strings
programming, using Fortran
 asa(1) — interprets ASA carriage control characters
 efl(1) — invokes the Extended Fortran Language
 f77(1) — invokes the Fortran 77 compiler
 fpr(1) — filters the output of Fortran programs for line printing
 fsplit(1) — splits f77 or efl files
programs, delaying running of
 sleep(1) — suspends the system for a specified interval of time
programs, establishing times for running
 at(1) — run commands at a later time
 crontab(1) — aids in the use of the cron process scheduling program
 cron(1M) — runs the clock daemon
programs, installation utilities
 cpset(1M) — installs files in specified directories
 finstall(1M) — installs A/UX software from specially prepared floppy
 disks
 install(1M) — places files in specified directories
programs, run-time environment settings
 env(1) — sets the environment for command execution
 nice(1) — executes a command at low priority
 nohup(1) — runs a command so that it can continue to run even after your
 session has ended
 shl(1) — manages the layering of multiple shells
 yes(1) — generates y entries in response to requests for input
 chroot(1M) — changes the root directory for a command
programs, running Macintosh applications
 launch(1) — runs a Macintosh binary application in A/UX
progress bar
 StartMonitor(1M) — displays a progress bar during the A/UX boot
 sequence
queues
 lpq(1) — queries the print spooler for progress information
 mailq(1M) — lists the contents of the mail queue
 msgctl(2) — message control operations
 msgget(2) — gets message queue

msgop(2) — message operations
 insque(3N) — insert/remove element from a queue
quiz
 quiz(6) — gives associative knowledge tests on various subjects
rain
 rain(6) — animates raindrops
random numbers
 drand48(3C) — generate uniformly distributed pseudo-random numbers
 rand(3C) — call a simple random-number generator
 rand(3F) — provide a Fortran uniform random-number generator
random text generation
 fortune(6) — plays the game of fortune telling
RCS
 ci(1) — checks in RCS revisions
 co(1) — checks out RCS revisions
 ident(1) — displays RCS keywords and their values
 merge(1) — merges three files into one
 rcs(1) — creates new RCS files or changes attributes of existing RCS files
 rcsdiff(1) — compares RCS revisions
 rcsintro(1) — introduces RCS commands
 rcsmerge(1) — merges two versions of an RCS file
 rlog(1) — displays log messages and other information about RCS files
 ucbsdifff3(1) — reports the differences between three files
 ucbsdifff(1) — reports differences between two files or directories
 sccstorcs(1M) — builds an RCS file from an SCCS file
 rcsfile(4) — format of an RCS file
reading files
 cat(1) — catenates and displays the contents of files
 head(1) — displays the first few lines of a file
 line(1) — reads one line from the standard input
 more(1) — show the contents of a file in display-size chunks
 pg(1) — shows the contents of a file in display-size chunks
 soelim(1) — eliminates the source commands from nroff input
 tail(1) — displays the last part of a file
 read(2) — reads from a file
 fread(3S) — produce binary input/output
 getc(3S) — get character or word from a stream
real group IDs
 getuid(2) — get real and effective user IDs and group IDs
 setregid(2) — sets real and effective group ID
real numbers
 aint(3F) — Fortran integer part intrinsic function

real user IDs

getuid(2) — get real and effective user IDs and group IDs

setreuid(2) — set real and effective user ID

setsid(2P) — create session and set process group ID

records, processing

colrm(1) — removes columns from a file

comm(1) — selects or rejects lines common to two sorted files

cut(1) — cuts out selected fields of each line of a file

join(1) — combines (joins) two relational files

paste(1) — merges lines of several files or subsequent lines of one file

sort(1) — sorts or merges files

uniq(1) — reports repeated lines in a file

redirection of output or input

cat(1) — catenates and displays the contents of files

csh(1) — runs the C shell, a command interpreter with C-like syntax

ksh(1) — runs the Korn shell, an enhanced command interpreter that is
backward-compatible with the Bourne shell (sh)

sh(1) — runs the Bourne shell

tee(1) — transcribes data

regular expressions

grep(1) — search a file for a specific pattern

regcmp(1) — compiles regular expressions with a file

regcmp(3X) — compile and execute a regular expression

regexp(5) — regular expression compile and match routines

relational joining of files

join(1) — combines (joins) two relational files

relocation

reloc(4) — relocation information for a common object file

remainders

floor(3M) — floor, ceiling, remainder, absolute value functions

mod(3F) — provide Fortran remaindering intrinsic functions

reminder service

calendar(1) — provides a reminder service

leave(1) — reminds you when you have to leave

Remote Procedure Call

rup(1N) — displays the status of machines on the local network (RPC
version)

rusers(1N) — produces a login list for local machines (RPC version)

nfsstat(1M) — displays Network File System (NFS) statistics

portmap(1M) — converts RPC program numbers into DARPA protocol
port numbers

rpcinfo(1M) — reports RPC information

rusersd(1M) — rusers invokes a server for users

spray(1M) — sprays packets

sprayd(1M) — returns information for the spray command
getrpcent(3N) — get RPC entry
getrpcport(3N) — gets a Remote Procedure Call (RPC) port number
rpc(3N) — library routines for remote procedure calls
spray(3N) — scatters data in order to check the network
rpc(4) — RPC program number database

remote systems

atprint(1) — transfers data to a printer by using AppleTalk protocols
ct(1C) — runs login on a dial-up line
cu(1C) — establishes an interactive connection with another system
rcp(1C) — copies files between two systems
rdist(1) — distributes remote files
remsh(1N) — invokes to a shell on a remote system
rlogin(1N) — logs in to a remote system
rup(1N) — displays the status of machines on the local network (RPC version)
rusers(1N) — produces a login list for local machines (RPC version)
tip(1C) — establishes a connection to a remote system
uucp(1C) — copies files from one system to another system
uname(1C) — displays the names of systems to which uucp and cu can connect
uusend(1C) — sends a file to a remote host
remlogin(1M) — runs on a remote system to log you in
remshd(1M) — invokes the remote shell server
restore(1M) — retrieve files from within a dump.bsd archive into an existing file system
rexecd(1M) — server for remote executions
rlogind(1M) — server for remote logins
rusersd(1M) — rusers invokes a server for users
showmount(1M) — shows all remote mounts
talkd(1M) — invokes the remote user communication server
uuxqt(1M) — handles requests from remote systems to run commands
mount(3N) — keeps track of remotely mounted file systems
rcmd(3N) — routines for returning a stream to a remote command
rexec(3N) — returns a stream to a remote command
rusers(3N) — return information about users on remote machines
rpc(3N) — library routines for remote procedure calls
rstat(3N) — get performance data from remote kernel
rttime(3) — gets remote time
rwall(3N) — writes to specified remote machines
sm_inter(3N) — status monitor protocol
xdr(3N) — provide library routines for external data representation
phones(4) — remote host telephone number database
remote(4) — remote host description file

rhosts(4N) — trusted hosts file format
rmtab(4) — remotely mounted file system table

removing

cancel(1) — cancels print requests spooled through the lp command
colrm(1) — removes columns from a file
cut(1) — cuts out selected fields of each line of a file
deroff(1) — removes nroff/troff, tbl, and eqn constructs
ipcrm(1) — removes interprocess communications facilities
kill(1) — terminates a process
lprm(1) — removes jobs from the line printer spooling queue for a
 Berkeley file system (4.2)
rm(1) — remove files or directories
rmdel(1) — removes a delta from an SCCS file
dev_kill(1M) — removes device files from a directory
killall(1M) — kills all active processes
flock(2) — applies or removes an advisory lock on an open file
rmdir(2) — remove a directory file
unlink(2) — remove directory entry
unmount(2) — remove a file system
insque(3N) — insert/remove element from a queue

repairing file systems

clri(1M) — clears inodes
fsck(1M) — checks file-system consistency and interactively repairs the
 file system
fsdb(1M) — debugs the file system
ncheck(1M) — locates the filename associated with an i-number
esch(8) — validates and repairs file systems from the A/UX Startup shell

repeated lines in text

uniq(1) — reports repeated lines in a file

resources, Macintosh

derez(1) — decompiles a resource file
fcnvt(1) — converts a file in one storage format to a different storage
 format
rez(1) — compiles Macintosh resource files from source code
setfile(1) — sets attributes for Macintosh files, such as file type and
 creator

reversing characters within lines

rev(1) — reverses characters within each line of text

Revision Control System

ci(1) — checks in RCS revisions
co(1) — checks out RCS revisions
ident(1) — displays RCS keywords and their values
merge(1) — merges three files into one
rcs(1) — creates new RCS files or changes attributes of existing RCS files

rcsdiff(1) — compares RCS revisions
rcsintr(1) — introduces RCS commands
rcsmmerge(1) — merges two versions of an RCS file
rlog(1) — displays log messages and other information about RCS files
ucbdiff3(1) — reports the differences between three files
ucbdiff(1) — reports differences between two files or directories
sccstorcs(1M) — builds an RCS file from an SCCS file
rcsfile(4) — format of an RCS file

robots

autorobots(6) — plays the game of autorobots
chase(6) — plays the game of chase
robots(6) — plays the game of robots

root directory

chroot(1M) — changes the root directory for a command
chroot(2) — changes the root directory

rounding

round(3F) — provide Fortran nearest integer functions

routing tables

route(1M) — manipulates the routing tables
routed(1M) — invokes the network routing daemon

RPC

rup(1N) — displays the status of machines on the local network (RPC version)
rusers(1N) — produces a login list for local machines (RPC version)
nfsstat(1M) — displays Network File System (NFS) statistics
portmap(1M) — converts RPC program numbers into DARPA protocol port numbers
rpcinfo(1M) — reports RPC information
rusersd(1M) — rusers invokes a server for users
spray(1M) — sprays packets
sprayd(1M) — returns information for the spray command
getrpcent(3N) — get RPC entry
getrpcport(3N) — gets a Remote Procedure Call (RPC) port number
rpc(3N) — library routines for remote procedure calls
spray(3N) — scatters data in order to check the network
rpc(4) — RPC program number database

run queue

lav(1) — displays load average statistics

running Macintosh applications from the command line

launch(1) — runs a Macintosh binary application in A/UX

SC40 Tape Backup

tcb(1) — blocks data to 8K for direct input to /dev/rmt/tcx
tc(7) — tape device driver

SCCS

- admin(1) — creates and administers SCCS files
- cdc(1) — changes the delta commentary of an SCCS delta
- comb(1) — combines SCCS deltas
- delta(1) — makes a delta (change) to an SCCS file
- get(1) — gets a version of an SCCS file
- help(1) — provides help information about SCCS commands and messages
- prs(1) — displays information about an SCCS file
- rmdel(1) — removes a delta from an SCCS file
- sact(1) — displays who has checked a Source Code Control System (SCCS) file out for editing
- sccs(1) — performs SCCS subsystem commands
- sccsdiff(1) — compares two versions of an SCCS file
- unget(1) — undoes a previous get of an SCCS file
- val(1) — validate SCCS file
- vc(1) — manipulates version control information inside a data stream
- what(1) — reports identification information for a file
- sccstorcs(1M) — builds an RCS file from an SCCS file
- sccsfile(4) — format of an SCCS file

SCCS deltas

- cdc(1) — changes the delta commentary of an SCCS delta
- comb(1) — combines SCCS deltas
- delta(1) — makes a delta (change) to an SCCS file
- rmdel(1) — removes a delta from an SCCS file
- sact(1) — displays who has checked a Source Code Control System (SCCS) file out for editing

screen management

- clear(1) — clears the terminal screen
- col(1) — filters text containing printer control sequences for use at a display device
- colcrt(1) — filters `nroff` output for terminal previewing
- ul(1) — filters special underlining sequences imbedded in text for use at a display device
- curses5.0(3X) — provides BSD-style screen functions with optimal cursor motion
- curses(3X) — CRT screen handling and optimization package

screen processing

- 300(1) — filter text containing printer control sequences for a DASI terminal
- 4014(1) — filters text containing printer control sequences a page at a time
- 450(1) — filters text containing printer control sequences for the DASI terminal

`tc(1)` — interprets troff output for use at a vintage display device

searching

`grep(1)` — search a file for a specific pattern

`bsearch(3C)` — performs a binary search on a sorted table

`hsearch(3C)` — manage hash search tables

`lsearch(3C)` — provide a linear search and update

`tsearch(3C)` — manage binary search trees

searching text

`freq(1)` — reports character frequencies in a file

`grep(1)` — search a file for a specific pattern

`lookbib(1)` — finds references in a bibliography

`wc(1)` — counts characters, words, and lines in a file

security

`login(1)` — signs you on a terminal session

`logname(1)` — gets the login name

`newgrp(1)` — logs you into a new group

`passwd(1)` — changes the login password

`rlogin(1N)` — logs in to a remote system

`Login(1M)` — logs you in to A/UX by using a graphical user interface

`remlogin(1M)` — runs on a remote system to log you in

`rlogind(1M)` — server for remote logins

`getlogin(3C)` — gets login name

`logname(3X)` — return login name of user

`auxstartuprc(4)` — authorization file that helps password-protect and otherwise secure A/UX Startup

`issue(4)` — project identification file format

`passwd(4)` — password file

`profile(4)` — setting up an environment at login time

segments

`brk(2)` — change data segment space allocation

`end(3C)` — last locations in program

`a.out(4)` — common assembler and link editor output

semaphores

`ipcrm(1)` — removes interprocess communications facilities

`ipcs(1)` — reports interprocess communication facilities status

`semctl(2)` — semaphore control operations

`semget(2)` — get set of semaphores

`semop(2)` — performs semaphore operations

serial communications

`ct(1C)` — runs `login` on a dial-up line

`cu(1C)` — establishes an interactive connection with another system

`kermit(1C)` — invokes the Kermit file-transfer program

`stty(1)` — sets the modes for a terminal

`tip(1C)` — establishes a connection to a remote system

tty(1) — obtains the device filename for the terminal or CommandShell window where it is invoked
updater(1) — updates files between two machines
uucp(1C) — copies files from one system to another system
uuencode(1C) — encode and decode a binary file
uusend(1C) — sends a file to a remote host
uustat(1C) — controls uucp jobs and provides status information
uuto(1C) — provide an easy interface to the uucp command, using the public directories
uux(1C) — runs a command on a remote system
getty(1M) — set the initial communication modes, such as speed and line discipline, for the purpose of logging users in to A/UX through serial lines
setport(1M) — sets the characteristics of a serial port
slattach(1M) — attaches a serial line to a network interface
slattconf(1M) — attaches a serial line to a network interface and configures the network interface
slip(1M) — assigns a serial line to a network interface
gettydefs(4) — speed and terminal settings used by getty
inittab(4) — script for the init process
ttytype(4) — database of terminal types by port
serial(7) — provides the on-board serial ports

Serial Line Internet Protocol

dslipuser(1M) — displays the current state of the Compressed Serial Line/Internet Protocol (CSL/IP) database
mkslipuser(1M) — creates or updates the Compressed Serial Line/Internet Protocol (CSL/IP) database
slip(1M) — assigns a serial line to a network interface
slip.config(4) — establishes the number of available Compressed Serial Line/Internet Protocol (CSL/IP) connections
slip.hosts(4) — maps login names to Compressed Serial Line/Internet Protocol (CSL/IP) client host names
slip.user(4) — database of available Compressed Serial Line/Internet Protocol (CSL/IP) connections

servers

ypwhich(1) — displays the host name of a system's Network Information Service (NIS) server
comsat(1M) — invokes the server for biff
exportfs(1M) — exports and unexports directories to Network File System (NFS) clients
fingerd(1M) — handles requests from remote systems for user information from finger
ftpd(1M) — provide Internet File Transfer Protocol (FTP) service
inetd(1M) — starts Internet servers when needed

mountd(1M) — invokes the Network File System (NFS) mount-request server
named(1M) — provides Internet domain name service
portmap(1M) — converts RPC program numbers into DARPA protocol port numbers
remshd(1M) — invokes the remote shell server
rexeed(1M) — server for remote executions
rlogind(1M) — server for remote logins
rstatd(1M) — invokes a server for kernel statistics
rusersd(1M) — rusers invokes a server for users
rwalld(1M) — invokes the network rwall server
rwhod(1M) — invokes the system status server
sprayd(1M) — returns information for the `spray` command
talkd(1M) — invokes the remote user communication server
telnetd(1M) — supports the DARPA standard TELNET protocol
tftpd(1M) — responds to requests to use the DARPA Trivial File Transfer Protocol
yppasswdd(1M) — handle requests to change a password served by the Network Information Service (NIS)
yppoll(1M) — reports the version of a Network Information Service (NIS) map that is on an NIS server
ypserv(1M) — provide Network Information Service (NIS) service
ypset(1M) — sets `ypbind` to a particular domain and Network Information Service (NIS) server
ypxfr(1M) — transfers a Network Information Service (NIS) map to the local system
servers(4) — Internet server database
slip.config(4) — establishes the number of available Compressed Serial Line/Internet Protocol (CSL/IP) connections

services

getservent(3N) — get a service entry
services(4N) — service name database

session status

logname(1) — gets the login name
printenv(1) — displays the value of variables set in the current environment
ps(1) — reports process status
pwd(1) — prints the name of the working directory
tty(1) — obtains the device filename for the terminal or CommandShell window where it is invoked
whoami(1) — prints effective current user ID

session, terminal

`CommandShell(1)` — manages command-interpretation windows and moderates access to the A/UX console window
`chsh(1)` — changes the default login shell
`csh(1)` — runs the C shell, a command interpreter with C-like syntax
`ksh(1)` — runs the Korn shell, an enhanced command interpreter that is backward-compatible with the Bourne shell (`sh`)
`rlogin(1N)` — logs in to a remote system
`script(1)` — starts a shell that records terminal input and output
`sh(1)` — runs the Bourne shell
`shl(1)` — manages the layering of multiple shells
`telnet(1C)` — communicates with another host via the TELNET protocol
`Login(1M)` — logs you in to A/UX by using a graphical user interface
`vt102(7)` — provides protocols for VT102 terminals

session, user interface preferences

`CommandShell(1)` — manages command-interpretation windows and moderates access to the A/UX console window
`chsh(1)` — changes the default login shell
`Login(1M)` — logs you in to A/UX by using a graphical user interface
`vt102(7)` — provides protocols for VT102 terminals

shared memory

`mkshlib(1)` — creates a shared library
`shmctl(2)` — shared memory control operations
`shmget(2)` — get shared memory segment
`shmop(2)` — shared memory operations

shared strings

`xstr(1)` — reports strings from C programs to implement shared strings

shell programming, boolean operations

`test(1)` — evaluates conditions
`true(1)` — provides truth values

shell programming, expression evaluation

`basename(1)` — get part of a pathname
`echo(1)` — echoes its arguments
`expr(1)` — evaluates arguments as an expression
`getopt(1)` — parses command options
`rev(1)` — reverses characters within each line of text

shell programming, input and output operations

`line(1)` — reads one line from the standard input
`query(1)` — queries the user for input
`tput(1)` — queries `terminfo` database
`macquery(1M)` — posts a Macintosh alert box to query the user
`vt102(7)` — provides protocols for VT102 terminals

shells

chsh(1) — changes the default login shell
csh(1) — runs the C shell, a command interpreter with C-like syntax
ksh(1) — runs the Korn shell, an enhanced command interpreter that is backward-compatible with the Bourne shell (sh)
remsh(1N) — invokes to a shell on a remote system
sh(1) — runs the Bourne shell
shl(1) — manages the layering of multiple shells
remshd(1M) — invokes the remote shell server
getusershell(3) — generate authenticated pathnames corresponding to executable shell programs
shells(4) — shell pathnames file
StartupShell(8) — interprets command lines such as those used to boot A/UX and check file systems within the A/UX Startup application

shutdown

powerdown(1M) — turns off power to the computer
shutdown(1M) — terminates processes that support multi-user mode and enters single-user mode

sign, numeric

sign(3F) — returns Fortran transfer-of-sign intrinsic functions

signal stack

sigstack(2) — set or get signal stack context

signals

ipcrm(1) — removes interprocess communications facilities
kill(1) — terminates a process
kill(2) — sends a signal to a process or a group of processes
pause(2) — suspends a process until signal
sigblock(2) — block signals
sigpause(2) — release blocked signals and wait for interrupt
sigpending(2P) — examine pending signals
sigsetmask(2) — set current signal mask
sigstack(2) — set or get signal stack context
sigvec(2) — optional BSD-compatible software signal facilities
killpg(3N) — sends signal to a process group
set42sig(3) — sets the Berkeley Software Distribution (BSD) 4.2 signal interface
sigaction(3P) — examine or change signal action
signal(3) — specifies what to do upon receipt of a signal
signal(3F) — specifies Fortran action on receipt of a system signal
sigprocmask(3P) — examines and changes blocked signals
sigsetops(3P) — manipulate signal sets
sigsuspend(3P) — waits for a signal
ssignal(3C) — produce software signals

sine

- `sin(3F)` — provide Fortran sine intrinsic functions
- `sinh(3F)` — provide Fortran hyperbolic sine intrinsic function
- `sinh(3M)` — provide hyperbolic functions
- `trig(3M)` — provide trigonometric functions

single-spaced text

- `ssp(1)` — produces single spaced output

SL/IP

- `dslipuser(1M)` — displays the current state of the Compressed Serial Line/Internet Protocol (CSL/IP) database
- `mkslipuser(1M)` — creates or updates the Compressed Serial Line/Internet Protocol (CSL/IP) database
- `slip(1M)` — assigns a serial line to a network interface
- `slip.config(4)` — establishes the number of available Compressed Serial Line/Internet Protocol (CSL/IP) connections
- `slip.hosts(4)` — maps login names to Compressed Serial Line/Internet Protocol (CSL/IP) client host names
- `slip.user(4)` — database of available Compressed Serial Line/Internet Protocol (CSL/IP) connections

slides

- `mvt(1)` — typeset documents that contain `troff` and `mm` or `mv` macro-formatting requests
- `mv(5)` — a `troff` macro package for typesetting viewgraphs and slides

SNOBOL

- `sno(1)` — runs the SNOBOL interpreter

SNOBOL programming

- `sno(1)` — runs the SNOBOL interpreter

".so" macro

- `soelim(1)` — eliminates the source commands from `nroff` input

sockets

- `accept(2N)` — accept a connection on a socket
- `bind(2N)` — bind a name to a socket
- `connect(2N)` — initiates a connection on a socket
- `getpeername(2N)` — gets the name of a connected peer
- `getsockname(2N)` — gets a socket name
- `getsockopt(2N)` — get and set options on sockets
- `listen(2N)` — listens for connections on a socket
- `recv(2N)` — receive a message from a socket
- `send(2N)` — send a message from a socket
- `shutdown(2N)` — shut down part of a full-duplex connection
- `socket(2N)` — create an endpoint for communication
- `socketpair(2)` — creates a pair of connected sockets
- `appletalk(7)` — interfaces with the AppleTalk protocols

software loopback

lo(5) — software loopback network interface

sorting

lorder(1) — finds the ordering relation for an object library

sort(1) — sorts or merges files

sortbib(1) — sorts bibliographic database

tsort(1) — sorts lines in a file topologically

qsort(3C) — performs a quicker sort

Source Code Control System

admin(1) — creates and administers SCCS files

cdc(1) — changes the delta commentary of an SCCS delta

comb(1) — combines SCCS deltas

delta(1) — makes a delta (change) to an SCCS file

get(1) — gets a version of an SCCS file

help(1) — provides help information about SCCS commands and messages

prs(1) — displays information about an SCCS file

rmdel(1) — removes a delta from an SCCS file

sact(1) — displays who has checked a Source Code Control System (SCCS) file out for editing

sccs(1) — performs SCCS subsystem commands

sccsdiff(1) — compares two versions of an SCCS file

unget(1) — undoes a previous get of an SCCS file

val(1) — validate SCCS file

what(1) — reports identification information for a file

sccstorcs(1M) — builds an RCS file from an SCCS file

sccsfile(4) — format of an SCCS file

source text management

admin(1) — creates and administers SCCS files

cb(1) — improves spacing and indentation of C source files

cdc(1) — changes the delta commentary of an SCCS delta

ci(1) — checks in RCS revisions

co(1) — checks out RCS revisions

comb(1) — combines SCCS deltas

get(1) — gets a version of an SCCS file

help(1) — provides help information about SCCS commands and messages

ident(1) — displays RCS keywords and their values

indent(1) — indents and formats C program source

lint(1) — invokes a C program checker

make(1) — maintains, updates, and regenerates groups of files

prs(1) — displays information about an SCCS file

rcs(1) — creates new RCS files or changes attributes of existing RCS files
 rcsdiff(1) — compares RCS revisions
 rcsintro(1) — introduces RCS commands
 rcsmerge(1) — merges two versions of an RCS file
 rlog(1) — displays log messages and other information about RCS files
 rmdel(1) — removes a delta from an SCCS file
 sact(1) — displays who has checked a Source Code Control System (SCCS) file out for editing
 sccs(1) — performs SCCS subsystem commands
 sccsdiff(1) — compares two versions of an SCCS file
 ucbsdifff3(1) — reports the differences between three files
 ucbsdifff(1) — reports differences between two files or directories
 unget(1) — undoes a previous get of an SCCS file
 val(1) — validate SCCS file
 what(1) — reports identification information for a file
 sccstorcs(1M) — builds an RCS file from an SCCS file
 rcsfile(4) — format of an RCS file
 sccsfile(4) — format of an SCCS file

spaces (in text)

expand(1) — expand tabs to spaces, and vice versa

spelling

spell(1) — find spelling errors

spline curves

spline(1G) — interpolates a smooth curve

spooler management

cancel(1) — cancels print requests spooled through the lp command
 enable(1) — enable or disable LP printers
 lpq(1) — queries the print spooler for progress information
 lpr(1) — spools print requests to printers
 lprm(1) — removes jobs from the line printer spooling queue for a Berkeley file system (4.2)
 lpstat(1) — prints lp status information
 uustat(1C) — controls uucp jobs and provides status information
 accept(1M) — allows lp requests
 lpadmin(1M) — configures the lp spooling system
 lpc(1M) — controls the operation of the line printer
 lpd(1M) — supports the Berkeley print spooler ." 4.2 line-printer daemon
 lpsched(1M) — start or stop the lp request scheduler and move requests
 lptest(1M) — generates a line-printer ripple pattern
 reject(1M) — prevents LP requests
 transcript(1M) — filter data for the POSTSCRIPT printers
 uucleanup(1M) — removes old files from the uucp spool directory

spraying

- spray(1M) — sprays packets
- sprayd(1M) — returns information for the spray command
- spray(3N) — scatters data in order to check the network

square root

- exp(3F) — Fortran exponential intrinsic function
- exp(3M) — provide exponential, logarithm, power, and square root functions
- sqrt(3F) — provide Fortran square root intrinsic functions

standard units

- units(1) — rescales quantities according to a the unit of measure specified

Star Trek

- trek(6) — plays the game of trek

startup

- login(1) — signs you on a terminal session
- newgrp(1) — logs you into a new group
- Login(1M) — logs you in to A/UX by using a graphical user interface
- StartMonitor(1M) — displays a progress bar during the A/UX boot sequence
- brc(1M) — execute system initialization shell scripts
- init(1M) — spawn general processes
- killall(1M) — kills all active processes
- powerdown(1M) — turns off power to the computer
- reboot(1M) — reboots the operating system
- shutdown(1M) — terminates processes that support multi-user mode and enters single-user mode
- startmsg(1M) — sends messages to StartMonitor, which displays a progress bar during the A/UX boot process
- startup(1M) — runs startup programs at boot time
- reboot(2) — reboot system or halt processor
- auxstartuprc(4) — authorization file that helps password-protect and otherwise secure A/UX Startup
- inittab(4) — script for the init process
- StartupShell(8) — interprets command lines such as those used to boot A/UX and check file systems within the A/UX Startup application
- intro(8) — introduces commands executed from the A/UX Startup shell

statistics

- lav(1) — displays load average statistics
- ff(1M) — lists file names and statistics for a System V file system
- ncstats(1M) — displays kernel name cache statistics
- nfsstat(1M) — displays Network File System (NFS) statistics
- rstatd(1M) — invokes a server for kernel statistics

statfs(2) — gets file-system statistics

ustat(2) — gets file system statistics

status

hostname(1N) — sets or displays the name of the current host system

last(1) — displays login and logout times for each user of the system

lpq(1) — queries the print spooler for progress information

lpstat(1) — prints lp status information

netstat(1N) — displays network status information

ps(1) — reports process status

ruptime(1N) — displays the host status of local machines

tty(1) — obtains the device filename for the terminal or CommandShell window where it is invoked

uptime(1) — reports how long system has been up

users(1) — reports a list of the users who are logged on to the system

w(1) — displays a summary of the current system activity

who(1) — reports users who are currently logged in to the system

whoami(1) — prints effective current user ID

mount(1M) — mount and unmount file systems

pstat(1M) — prints system facts

rwod(1M) — invokes the system status server

showmount(1M) — shows all remote mounts

whodo(1M) — informs you of the current system activity

rstat(3N) — get performance data from remote kernel

rtime(3) — gets remote time

status, file system

df(1) — reports the used and unused storage capacity for a file system

du(1) — summarizes disk usage

status, session

logname(1) — gets the login name

printenv(1) — displays the value of variables set in the current environment

ps(1) — reports process status

pwd(1) — prints the name of the working directory

tty(1) — obtains the device filename for the terminal or CommandShell window where it is invoked

whoami(1) — prints effective current user ID

status, system

finger(1) — displays information about the users on a system

groups(1) — displays group memberships

hostid(1N) — sets or displays the identifier of the current host system

hostname(1N) — sets or displays the name of the current host system

id(1) — displays user and group IDs and names

last(1) — displays login and logout times for each user of the system

machid(1) — provide truth values about processor type

pagesize(1) — displays the system page size
uname(1) — displays identification information about the current system
uptime(1) — reports how long system has been up
users(1) — reports a list of the users who are logged on to the system
w(1) — displays a summary of the current system activity
who(1) — reports users who are currently logged in to the system
whoDo(1M) — informs you of the current system activity

streams (data)

line_sane(1M) — pushes streams line disciplines
fclose(3S) — close or flush a stream
ferror(3S) — stream status inquiries
fopen(3S) — open a stream
fread(3S) — produce binary input/output
fseek(3S) — reposition a file pointer in a stream
getc(3S) — get character or word from a stream
gets(3S) — get a string from a stream
line_push(3) — routine used to push streams line disciplines
printf(3S) — format and output string and numeric data
putc(3S) — put a character or word on a stream
puts(3S) — put a string on a stream
rcmd(3N) — routines for returning a stream to a remote command
rexec(3N) — returns a stream to a remote command
scanf(3S) — convert formatted input
setbuf(3S) — assign buffering to a stream
ungetc(3S) — pushes a character back into input stream
streams(7) — provides an interface for character I/O

strings

basename(1) — get part of a pathname
grep(1) — search a file for a specific pattern
rev(1) — reverses characters within each line of text
strings(1) — finds the printable strings in an object or other binary file
xstr(1) — reports strings from C programs to implement shared strings
atof(3C) — converts an ASCII string to floating-point number
bstring(3) — bit and byte string operations
ecvt(3C) — convert floating-point number to string
gets(3S) — get a string from a stream
index(3F) — return location of Fortran substring
len(3F) — return length of Fortran string
lge(3F) — string comparison intrinsic functions
puts(3S) — put a string on a stream
string(3C) — provide string operations
strtod(3C) — converts a string to a double-precision number
strtoul(3C) — convert strings to integer

subroutines

intro(3) — introduces the subroutines and libraries

subtraction

dim(3F) — Fortran positive difference intrinsic functions

superblock

sync(1) — updates the superblock

fsck(1M) — checks file-system consistency and interactively repairs the file system

mkfs(1M) — constructs a System V file system

sync(2) — update superblock

inode(4) — format of a System V inode

svfs(4) — System V system volume format

ufs(4) — UFS file-system format

suspend execution

shl(1) — manages the layering of multiple shells

sleep(1) — suspends the system for a specified interval of time

sigpause(2) — release blocked signals and wait for interrupt

wait3(2N) — wait for child process to stop or terminate

wait(2) — wait for child process to stop or terminate

sigsuspend(3P) — waits for a signal

sleep(3C) — suspends execution for interval

tcdrain(3P) — provide line control functions

usleep(3) — suspend execution for interval

SVFS

mkfs(1M) — constructs a System V file system

dir(4) — format of System V directories

inode(4) — format of a System V inode

svfs(4) — System V system volume format

swapping (memory)

swap(1M) — adds disk blocks to or deletes them from the swap area

swab(3C) — swaps bytes

symbol table

cc(1) — invokes the C compiler

ld(1) — invokes the link editor for common object files

nm(1) — displays the symbol table of a common object file

strip(1) — strips symbol and line number information from an object file

ldgetname(3X) — retrieves symbol name for object file symbol table entry

ldtbindex(3X) — compute index of a symbol table entry of a common object file

ldtbread(3X) — read an indexed symbol table entry of a common object file

ldtbseek(3X) — seek to the symbol table of a common object file

nlist(3C) — gets entries from name list

syms(4) — common object file symbol table format

synchronization

select(2N) — synchronous I/O multiplexing

system activity

ipcs(1) — reports interprocess communication facilities status

lav(1) — displays load average statistics

ps(1) — reports process status

sag(1G) — generates a system activity graph

sar(1) — reports system activity

sysline(1) — displays the system status on the status line of a terminal

timex(1) — reports the elapsed, user, and system time during the execution of a command

w(1) — displays a summary of the current system activity

acct(1M) — present an overview of accounting commands

acctcms(1M) — summarizes commands from per-process accounting records

acctcom(1M) — searches and formats process accounting files

acctcon(1M) — invoke connect-time accounting

acctmerg(1M) — merges or adds accounting files

acctprc(1M) — provide process accounting

acctsh(1M) — provide shell procedures for accounting

diskusg(1M) — generates disk accounting data by user ID

fwtmp(1M) — manipulate connect accounting records

pac(1M) — gathers printer/plotter accounting information

runacct(1M) — runs daily accounting

sadc(1M) — report system activity

whodo(1M) — informs you of the current system activity

system administration, backing up file systems

bcopy(1M) — copies blocks interactively

dcopy(1M) — copies System V File System-style file systems for optimal access time

dump.bsd(1M) — create a dump.bsd archive by making copies of files from a given file system

escher(1M) — helps you with autorecovery administration

eu(1M) — updates autorecovery files

eupdate(1M) — updates important files for autorecovery purposes

finc(1M) — generates a fast incremental backup for System V file systems

frec(1M) — recovers files from a backup tape

restore(1M) — retrieve files from within a dump.bsd archive into an existing file system

volcopy(1M) — copy file systems with label checking

system administration, file systems

`df(1)` — reports the used and unused storage capacity for a file system
`du(1)` — summarizes disk usage
`fstyp(1)` — reports the file-system type
`sync(1)` — updates the superblock
`clri(1M)` — clears inodes
`devnm(1M)` — displays the current device name
`ff(1M)` — lists file names and statistics for a System V file system
`fsck(1M)` — checks file-system consistency and interactively repairs the file system
`fsdb(1M)` — debugs the file system
`fsentry(1M)` — creates an entry in the file-system table
`fsirand(1M)` — installs random inode generation numbers
`fsstat(1M)` — reports the state of a file system
`fuser(1M)` — identifies processes using a file or file structure
`mkfs1b(1M)` — constructs a file system with 512-byte blocks
`mkfs(1M)` — constructs a System V file system
`mklost+found(1M)` — makes a directory named `lost+found` to be used by `fsck`
`mount(1M)` — mount and unmount file systems
`ncheck(1M)` — locates the filename associated with an i-number
`newfs(1M)` — makes a Berkeley 4.2 (UFS) file system
`tunefs(1M)` — tunes a Berkeley 4.2 (UFS) file system

system administration, general

`checkinstall(1)` — checks the installation of boards
`tset(1)` — set or reset the terminal to a sensible state
`badblk(1M)` — sets or updates bad block information
`chgnod(1M)` — changes the current A/UX system node name
`diskformat(1M)` — formats a disk through a driver-dependent format operation
`dp(1M)` — performs disk partitioning
`getty(1M)` — set the initial communication modes, such as speed and line discipline, for the purpose of logging users in to A/UX through serial lines
`line_sane(1M)` — pushes streams line disciplines
`pname(1M)` — associates named partitions with device files
`setport(1M)` — sets the characteristics of a serial port
`settimezone(1M)` — sets the local time zone
`swap(1M)` — adds disk blocks to or deletes them from the swap area
`tic(1M)` — compiles (translates) `terminfo` files
`tty_add(1M)` — modify the `/etc/inittab` file in terms of enabling serial ports for use as login terminals
`tzdump(1M)` — displays the date and time for one or more time zones
`tzic(1M)` — compiles time-zone information files that are required to set

the local time-zone

system administration, installing software

- cpset(1M) — installs files in specified directories
- finstall(1M) — installs A/UX software from specially prepared floppy disks
- install(1M) — places files in specified directories

system administration, kernel

- autoconfig(1M) — creates an up-to-date kernel
- kconfig(1M) — tunes kernel parameters for work-load optimization
- module_dump(1M) — queries kernel files for configuration information
- newconfig(1M) — generates an up-to-date kernel
- newunix(1M) — manipulates the files that determine the configuration of a new kernel

system administration, mail

- rmail(1) — handles remote mail received via UUCP
- comsat(1M) — invokes the server for biff
- mailq(1M) — lists the contents of the mail queue
- newaliases(1M) — rebuilds the database for the mail aliases file
- sendmail(1M) — sends mail

system administration, NFS file systems

- domainname(1) — sets or displays the name of the Network Information Service (NIS) domain
- automount(1M) — mounts Network File System (NFS) when needed
- exportfs(1M) — exports and unexports directories to Network File System (NFS) clients
- lockd(1M) — handle local and remote lock requests
- mountd(1M) — invokes the Network File System (NFS) mount-request server
- nfsd(1M) — invoke the NFS daemons
- nfsstat(1M) — displays Network File System (NFS) statistics
- rpcinfo(1M) — reports RPC information
- showmount(1M) — shows all remote mounts
- spray(1M) — sprays packets
- sprayd(1M) — returns information for the spray command
- statd(1M) — provide crash and recovery monitoring for network locking services

system administration, spoolers

- enable(1) — enable or disable LP printers
- lpstat(1) — prints lp status information
- accept(1M) — allows lp requests
- lpadmin(1M) — configures the lp spooling system
- lpc(1M) — controls the operation of the line printer
- lpd(1M) — supports the Berkeley print spooler . " 4.2 line-printer daemon
- lpsched(1M) — start or stop the lp request scheduler and move requests

lpctest(1M) — generates a line-printer ripple pattern
reject(1M) — prevents LP requests
transcript(1M) — filter data for the POSTSCRIPT printers

system administration, user accounts

chfn(1) — changes the real-name field of your password file entry for use
by finger
chsh(1) — changes the default login shell
finger(1) — displays information about the users on a system
adduser(1M) — adds a user account
fingerd(1M) — handles requests from remote systems for user
information from finger
pwck(1M) — check the password/group files
vipw(1M) — edits the password file

system administration, utilities for

su(1) — substitutes user ID
dev_kill(1M) — removes device files from a directory
mknod(1M) — builds a device file

system administration, UUCP

Uutry(1M) — contacts a remote system with debugging on
uucheck(1M) — checks the uucp directories and files
uucico(1M) — transfers files as specified by uucp work files
uucleanup(1M) — removes old files from the uucp spool directory
uucpd(1M) — handles the transfer of files by uucico over TCP/IP
connections
uudemon.admin(1M) — mails current uucp work status to the uucp
administrator
uudemon.cleanup(1M) — cleans up files in the uucp spool directory
uudemon.hour(1M) — processes spooled uucp requests
uudemon.poll(1M) — sets up polling for selected systems
uusched(1M) — schedules uucp file transfers

system calls

intro(2) — introduces system calls and error numbers
syscall(2) — indirect system call

system configuration

checkinstall(1) — checks the installation of boards
tset(1) — set or reset the terminal to a sensible state
adduser(1M) — adds a user account
autoconfig(1M) — creates an up-to-date kernel
badblk(1M) — sets or updates bad block information
chgnod(1M) — changes the current A/UX system node name
diskformat(1M) — formats a disk through a driver-dependent format
operation
dp(1M) — performs disk partitioning
getty(1M) — set the initial communication modes, such as speed and

line discipline, for the purpose of logging users in to A/UX through serial lines

`init(1M)` — spawn general processes

`kconfig(1M)` — tunes kernel parameters for work-load optimization

`line_sane(1M)` — pushes streams line disciplines

`lpadmin(1M)` — configures the `lp` spooling system

`module_dump(1M)` — queries kernel files for configuration information

`newconfig(1M)` — generates an up-to-date kernel

`newunix(1M)` — manipulates the files that determine the configuration of a new kernel

`pname(1M)` — associates named partitions with device files

`pstat(1M)` — prints system facts

`setport(1M)` — sets the characteristics of a serial port

`settimezone(1M)` — sets the local time zone

`slattconf(1M)` — attaches a serial line to a network interface and configures the network interface

`swap(1M)` — adds disk blocks to or deletes them from the swap area

`tic(1M)` — compiles (translates) `terminfo` files

`tty_add(1M)` — modify the `/etc/inittab` file in terms of enabling serial ports for use as login terminals

`tzdump(1M)` — displays the date and time for one or more time zones

`tzic(1M)` — compiles time-zone information files that are required to set the local time-zone

`uvar(2)` — returns system-specific configuration information

`gettydefs(4)` — speed and terminal settings used by `getty`

`inittab(4)` — script for the `init` process

`master(4)` — master kernel-configuration file format

system crashes

`errdead(1M)` — extracts error records from a crash dump

`statd(1M)` — provide crash and recovery monitoring for network locking services

system folder, personalizing

`systemfolder(1)` — create a personal System Folder

system kernel, generation of

`autoconfig(1M)` — creates an up-to-date kernel

`kconfig(1M)` — tunes kernel parameters for work-load optimization

`module_dump(1M)` — queries kernel files for configuration information

`newconfig(1M)` — generates an up-to-date kernel

`newunix(1M)` — manipulates the files that determine the configuration of a new kernel

system name

`hostname(1N)` — sets or displays the name of the current host system

`uname(1)` — displays identification information about the current system

`uname(2)` — get name of current system

HOSTNAME(4) — host name and domain name database

system startup and shutdown

StartMonitor(1M) — displays a progress bar during the A/UX boot sequence

brc(1M) — execute system initialization shell scripts

init(1M) — spawn general processes

killall(1M) — kills all active processes

powerdown(1M) — turns off power to the computer

reboot(1M) — reboots the operating system

shutdown(1M) — terminates processes that support multi-user mode and enters single-user mode

startmsg(1M) — sends messages to StartMonitor, which displays a progress bar during the A/UX boot process

startup(1M) — runs startup programs at boot time

system status

finger(1) — displays information about the users on a system

groups(1) — displays group memberships

hostid(1N) — sets or displays the identifier of the current host system

hostname(1N) — sets or displays the name of the current host system

id(1) — displays user and group IDs and names

last(1) — displays login and logout times for each user of the system

lpq(1) — queries the print spooler for progress information

lpstat(1) — prints lp status information

machid(1) — provide truth values about processor type

netstat(1N) — displays network status information

pagesize(1) — displays the system page size

ps(1) — reports process status

ruptime(1N) — displays the host status of local machines

tty(1) — obtains the device filename for the terminal or CommandShell window where it is invoked

uname(1) — displays identification information about the current system

uptime(1) — reports how long system has been up

users(1) — reports a list of the users who are logged on to the system

w(1) — displays a summary of the current system activity

who(1) — reports users who are currently logged in to the system

whoami(1) — prints effective current user ID

errdead(1M) — extracts error records from a crash dump

errdemon(1M) — calls the error-logging daemon

errpt(1M) — processes a report of logged errors

errstop(1M) — terminates the error-logging daemon

exterr(1M) — turns on/off the reporting of extended errors

mount(1M) — mount and unmount file systems

ncstats(1M) — displays kernel name cache statistics

pstat(1M) — prints system facts

`rwod(1M)` — invokes the system status server
`showmount(1M)` — shows all remote mounts
`whodo(1M)` — informs you of the current system activity

system time

`date(1)` — displays and sets the date
`settimezone(1M)` — sets the local time zone
`adjtime(2)` — correct the system time
`gettimeofday(2)` — get/set date and time
`time(2)` — get time

system variables

`kconfig(1M)` — tunes kernel parameters for work-load optimization
`sysconf(3P)` — gets configurable system variables

tables (in text)

`col(1)` — filters text containing printer control sequences for use at a display device
`deroff(1)` — removes `nroff/troff`, `tbl`, and `eqn` constructs
`tbl(1)` — table formatter for `nroff` or `troff`

tabs

`expand(1)` — expand tabs to spaces, and vice versa
`tabs(1)` — sets the tab stops on a terminal

tags

`ctags(1)` — maintains a tags file for a C program

tangent

`tan(3F)` — Fortran tangent intrinsic function
`tanh(3F)` — Fortran hyperbolic tangent intrinsic function
`trig(3M)` — provide trigonometric functions

tape (backup)

`cp(1)` — copies files
`cpio(1)` — copies files to or from a `cpio` archive
`pax(1)` — copies files to or from an archive in an IEEE format
`tar(1)` — copies files to or from a `tar` archive
`dump.bsd(1M)` — create a `dump.bsd` archive by making copies of files from a given file system
`finc(1M)` — generates a fast incremental backup for System V file systems
`freq(1M)` — recovers files from a backup tape
`restore(1M)` — retrieve files from within a `dump.bsd` archive into an existing file system
`dump.bsd(4)` — format of a file-system dump
`tc(7)` — tape device driver

tape drives

`mt(1)` — manipulates magnetic tape media
`tar(1)` — copies files to or from a `tar` archive
`tc(1)` — blocks data to 8K for direct input to `/dev/rmt/tcx`

tp(1) — copies files to or from a tp archive
tar(4) — format of tar header
mtio(7) — provides an interface library for magnetic tape devices
tc(7) — tape device driver

TCP

trpt(1M) — prints a readable description of TCP trace records

TCP/IP, maintenance of

netstat(1N) — displays network status information
nslookup(1) — interactively queries name servers
arp(1M) — displays and modifies the address translation table
dslipuser(1M) — displays the current state of the Compressed Serial
Line/Internet Protocol (CSL/IP) database
etheraddr(1M) — displays the Ethernet address of each Ethernet card
in your system
ftpd(1M) — provide Internet File Transfer Protocol (FTP) service
ifconfig(1M) — manages network interfaces
inetd(1M) — starts Internet servers when needed
mkslipuser(1M) — creates or updates the Compressed Serial
Line/Internet Protocol (CSL/IP) database
named(1M) — provides Internet domain name service
ping(1M) — exercises the TCP/IP network by sending Internet Control
Message Protocol (ICMP) packets to a named host
portmap(1M) — converts RPC program numbers into DARPA protocol
port numbers
remshd(1M) — invokes the remote shell server
rexecd(1M) — server for remote executions
rlogind(1M) — server for remote logins
route(1M) — manipulates the routing tables
routed(1M) — invokes the network routing daemon
rstatd(1M) — invokes a server for kernel statistics
rusersd(1M) — rusers invokes a server for users
rwalld(1M) — invokes the network rwall server
rwhod(1M) — invokes the system status server
slattach(1M) — attaches a serial line to a network interface
slattconf(1M) — attaches a serial line to a network interface and
configures the network interface
stdhosts(1M) — converts Internet addresses to standard form
talkd(1M) — invokes the remote user communication server
telnetd(1M) — supports the DARPA standard TELNET protocol
trpt(1M) — prints a readable description of TCP trace records

Tektronix 4014 terminal

4014(1) — filters text containing printer control sequences a page at a
time
tc(1) — interprets troff output for use at a vintage display device

Teletype Model 37

`greek(5)` — graphics for the extended TTY-37 type-box

teletype transmission

`tset(1)` — set or reset the terminal to a sensible state

TELNET

`telnet(1C)` — communicates with another host via the TELNET protocol

`telnetd(1M)` — supports the DARPA standard TELNET protocol

terminal capabilities

`300(1)` — filter text containing printer control sequences for a DASI terminal

`4014(1)` — filters text containing printer control sequences a page at a time

`450(1)` — filters text containing printer control sequences for the DASI terminal

`tput(1)` — queries `terminfo` database

`termcap(3X)` — provide terminal independent operation routines

`printcap(4)` — printer-capability database

`term(4)` — format of compiled term file

`termcap(4)` — terminal capability database

`terminfo(4)` — terminal capability database

terminal emulation

`CommandShell(1)` — manages command-interpretation windows and moderates access to the A/UX console window

`vt102(7)` — provides protocols for VT102 terminals

terminal modes

`stty(1)` — sets the modes for a terminal

`termio(7)` — provides a general terminal interface

terminal names

`term(5)` — conventional names for terminals

terminal screen

`clear(1)` — clears the terminal screen

`col(1)` — filters text containing printer control sequences for use at a display device

`colcrt(1)` — filters `nroff` output for terminal previewing

`ul(1)` — filters special underlining sequences imbedded in text for use at a display device

`curses5.0(3X)` — provides BSD-style screen functions with optimal cursor motion

`curses(3X)` — CRT screen handling and optimization package

terminal session

`CommandShell(1)` — manages command-interpretation windows and moderates access to the A/UX console window

`chsh(1)` — changes the default login shell

`csh(1)` — runs the C shell, a command interpreter with C-like syntax
`ksh(1)` — runs the Korn shell, an enhanced command interpreter that is backward-compatible with the Bourne shell (`sh`)
`rlogin(1N)` — logs in to a remote system
`script(1)` — starts a shell that records terminal input and output
`sh(1)` — runs the Bourne shell
`shl(1)` — manages the layering of multiple shells
`telnet(1C)` — communicates with another host via the TELNET protocol
`Login(1M)` — logs you in to A/UX by using a graphical user interface
`vt102(7)` — provides protocols for VT102 terminals

terminal settings

`stty(1)` — sets the modes for a terminal
`tabs(1)` — sets the tab stops on a terminal
`getty(1M)` — set the initial communication modes, such as speed and line discipline, for the purpose of logging users in to A/UX through serial lines
`keyset(1M)` — sets the keyboard for the console
`gettydefs(4)` — speed and terminal settings used by `getty`
`ioctl.syscon(4)` — console terminal settings file
`vt102(7)` — provides protocols for VT102 terminals

terminal types

`getty(1M)` — set the initial communication modes, such as speed and line discipline, for the purpose of logging users in to A/UX through serial lines
`termcap(3X)` — provide terminal independent operation routines
`termcap(4)` — terminal capability database
`terminfo(4)` — terminal capability database
`ttytype(4)` — database of terminal types by port
`vt102(7)` — provides protocols for VT102 terminals

terminals, general

`clear(1)` — clears the terminal screen
`ct(1C)` — runs `login` on a dial-up line
`greek(1)` — filters text for vintage display devices
`last(1)` — displays login and logout times for each user of the system
`stty(1)` — sets the modes for a terminal
`tabs(1)` — sets the tab stops on a terminal
`tset(1)` — set or reset the terminal to a sensible state
`tty(1)` — obtains the device filename for the terminal or CommandShell window where it is invoked
`tic(1M)` — compiles (translates) `terminfo` files
`tty_add(1M)` — modify the `/etc/inittab` file in terms of enabling serial ports for use as login terminals
`ctermid(3S)` — generate filename for terminal

dial(3C) — establishes an out-going terminal line connection
tcgetattr(3P) — get and set the terminal state
termcap(3X) — provide terminal independent operation routines
ttyname(3C) — find name of a terminal
nterm(5) — terminal driving tables for nroff
pty(7) — provides a pseudo terminal driver
termio(7) — provides a general terminal interface
termios(7P) — provides a A/UX® POSIX general terminal interface
tty(7) — controls the terminal interface
vt102(7) — provides protocols for VT102 terminals

termination, process

kill(1) — terminates a process
nohup(1) — runs a command so that it can continue to run even after your session has ended
killall(1M) — kills all active processes
shutdown(1M) — terminates processes that support multi-user mode and enters single-user mode
exit(2) — terminate process
abort(3C) — generates an IOT fault
abort(3F) — terminates a Fortran program

testing a line printer

lptest(1M) — generates a line-printer ripple pattern

testing a network

ping(1M) — exercises the TCP/IP network by sending Internet Control Message Protocol (ICMP) packets to a named host
lo(5) — software loopback network interface

text, checking embedded markups for errors

checkmm(1) — check documents formatted with the mm macros
checknr(1) — checks nroff/troff files
deroff(1) — removes nroff/troff, tbl, and eqn constructs
diffmk(1) — marks the differences between two files
hyphen(1) — finds hyphenated words
macref(1) — produces a cross-reference listing of macro files

text, editing

TextEditor(1) — lets you edit files interactively through mouse and menu operations
bfs(1) — edits big files
ed(1) — edit text
ex(1) — edit text
nl(1) — processes a file through a line numbering filter
ssp(1) — produces single spaced output
vi(1) — invokes the screen-oriented (visual) display editor

text, establishing fonts for troff typesetting

madev(1) — prepares `troff` description files

text, filtering out printer motions

300(1) — filter text containing printer control sequences for a DASI terminal

4014(1) — filters text containing printer control sequences a page at a time

450(1) — filters text containing printer control sequences for the DASI terminal

col(1) — filters text containing printer control sequences for use at a display device

colcrt(1) — filters `nroff` output for terminal previewing

greek(1) — filters text for vintage display devices

tc(1) — interprets `troff` output for use at a vintage display device

ul(1) — filters special underlining sequences imbedded in text for use at a display device

text, formatting and typesetting

daps(1) — invokes the Autologic APS-5 phototypesetter `troff` post-processor

enscript(1) — converts text files to `format` for printing

eqn(1) — format mathematical text for `troff`

fmt(1) — invokes a simple text formatter

fold(1) — folds long lines for finite-width output device

mm(1) — formats documents that contain `nroff` and `mm` macro formatting requests

mmt(1) — typeset documents that contain `troff` and `mm` or `mv` macro-formatting requests

mvt(1) — typeset documents that contain `troff` and `mm` or `mv` macro-formatting requests

neqn(1) — formats mathematical text for `nroff`

newform(1) — changes the format of a text file

nroff(1) — text formatter

otroff(1) — formats text for a specific phototypesetter

pr(1) — formats text for a print device

psdit(1) — converts `troff` intermediate format to POSTSCRIPT format

psroff(1) — formats a file through `troff` so it can be printed on a POSTSCRIPT printer

roffbib(1) — prints out all records in a bibliographic database

tbl(1) — table formatter for `nroff` or `troff`

troff(1) — formats and typesets files

text lines, filling and wrapping

fmt(1) — invokes a simple text formatter

fold(1) — folds long lines for finite-width output device

text lines, processing

`awk(1)` — scans a file for lines that match a specific pattern
`colrm(1)` — removes columns from a file
`comm(1)` — selects or rejects lines common to two sorted files
`cut(1)` — cuts out selected fields of each line of a file
`grep(1)` — search a file for a specific pattern
`head(1)` — displays the first few lines of a file
`join(1)` — combines (joins) two relational files
`line(1)` — reads one line from the standard input
`newform(1)` — changes the format of a text file
`nl(1)` — processes a file through a line numbering filter
`paste(1)` — merges lines of several files or subsequent lines of one file
`rev(1)` — reverses characters within each line of text
`sed(1)` — edits a stream of data
`sort(1)` — sorts or merges files
`tail(1)` — displays the last part of a file
`uniq(1)` — reports repeated lines in a file
`wc(1)` — counts characters, words, and lines in a file

text, preprocessing before formatting and typesetting

`cw(1)` — prepare constant-width text for `otroff`
`eqn(1)` — format mathematical text for `troff`
`grap(1)` — invokes a `pic` preprocessor for drawing graphs
`neqn(1)` — formats mathematical text for `nroff`
`pic(1)` — preprocesses `troff` files that contain drawings
`soelim(1)` — eliminates the source commands from `nroff` input
`tbl(1)` — table formatter for `nroff` or `troff`

text, processing of tabs within

`expand(1)` — expand tabs to spaces, and vice versa
`newform(1)` — changes the format of a text file

text processor

`awk(1)` — scans a file for lines that match a specific pattern
`col(1)` — filters text containing printer control sequences for use at a display device
`comm(1)` — selects or rejects lines common to two sorted files
`cpp(1)` — invokes the C language preprocessor
`daps(1)` — invokes the Autologic APS-5 phototypesetter `troff` post-processor
`deroff(1)` — removes `nroff/troff`, `tbl`, and `eqn` constructs
`eqn(1)` — format mathematical text for `troff`
`expand(1)` — expand tabs to spaces, and vice versa
`fmt(1)` — invokes a simple text formatter
`fold(1)` — folds long lines for finite-width output device
`grap(1)` — invokes a `pic` preprocessor for drawing graphs
`iw2(1)` — prepares data to be printed on the Apple ImageWriter II printer

- m4(1) — processes macros for C and other languages
- neqn(1) — formats mathematical text for nroff
- pic(1) — preprocesses troff files that contain drawings
- pr(1) — formats text for a print device
- rev(1) — reverses characters within each line of text
- sed(1) — edits a stream of data
- sort(1) — sorts or merges files
- ssp(1) — produces single spaced output
- tabs(1) — sets the tab stops on a terminal
- tbl(1) — table formatter for nroff or troff
- tr(1) — translates characters
- uniq(1) — reports repeated lines in a file
- text, searches**
 - freq(1) — reports character frequencies in a file
 - grep(1) — search a file for a specific pattern
 - lookbib(1) — finds references in a bibliography
 - wc(1) — counts characters, words, and lines in a file
- text, transforming**
 - awk(1) — scans a file for lines that match a specific pattern
 - m4(1) — processes macros for C and other languages
 - sed(1) — edits a stream of data
 - tr(1) — translates characters
- text, utilities for generating and spell checking**
 - addbib(1) — creates or extends a bibliographic database
 - diction(1) — locate wordy sentences in a document
 - indxbib(1) — builds an inverted index for a bibliography
 - ndx(1) — creates a subject-page index for a document
 - ptx(1) — generates a permuted index
 - refer(1) — finds and inserts literature references in documents
 - sortbib(1) — sorts bibliographic database
 - spell(1) — find spelling errors
 - style(1) — analyzes the surface characteristics of documents
 - subj(1) — generates a list of subjects from documents
- TFTP (Trivial File Transfer Protocol)**
 - tftp(1C) — transfers files via the Trivial File Transfer Protocol (TFTP)
 - tftpd(1M) — responds to requests to use the DARPA Trivial File Transfer Protocol
- three-byte integers**
 - l3tol(3C) — convert between 3-byte integers and long integers
- tic-tac-toe**
 - ttt(6) — play the game of tic-tac-toe

time and date

`cal(1)` — displays a calendar
`calendar(1)` — provides a reminder service
`date(1)` — displays and sets the date
`leave(1)` — reminds you when you have to leave
`cron(1M)` — runs the clock daemon
`settimeofday(1M)` — sets the local time zone
`gettimeofday(2)` — get/set date and time
`stime(2)` — set time
`time(2)` — get time
`ctime(3)` — convert date and time to ASCII
`tzfile(4)` — time-zone information
`nvrnm(7)` — provides an interface to nonvolatile memory

time zones

`settimeofday(1M)` — sets the local time zone
`tzdump(1M)` — displays the date and time for one or more time zones
`tzic(1M)` — compiles time-zone information files that are required to set the local time-zone
`tzfile(4)` — time-zone information

timers

`leave(1)` — reminds you when you have to leave
`getitimer(2)` — get/set value of interval timer

toolbox, Macintosh

`slots(3X)` — provides ROM library functions

topological sorting

`tsort(1)` — sorts lines in a file topologically

tracing

`trpt(1M)` — prints a readable description of TCP trace records
`ptrace(2)` — process trace

TranScript®

`transcript(1M)` — filter data for the POSTSCRIPT printers

transferring files

`cpio(1)` — copies files to or from a `cpio` archive
`cu(1C)` — establishes an interactive connection with another system
`ftp(1N)` — transfers files by using the DARPA Internet File Transfer Protocol (FTP)
`kermit(1C)` — invokes the Kermit file-transfer program
`pax(1)` — copies files to or from an archive in an IEEE format
`rcp(1C)` — copies files between two systems
`remsh(1N)` — invokes to a shell on a remote system
`tar(1)` — copies files to or from a `tar` archive
`tftp(1C)` — transfers files via the Trivial File Transfer Protocol (TFTP)
`tip(1C)` — establishes a connection to a remote system
`updater(1)` — updates files between two machines

uucp(1C) — copies files from one system to another system
uuencode(1C) — encode and decode a binary file
ftpd(1M) — provide Internet File Transfer Protocol (FTP) service
tftpd(1M) — responds to requests to use the DARPA Trivial File
Transfer Protocol
uucico(1M) — transfers files as specified by uucp work files

translators

tr(1) — translates characters
uuencode(1C) — encode and decode a binary file
conv(3C) — translate characters
number(6) — converts Arabic numerals to English

Transliterate Protocol Trace

trpt(1M) — prints a readable description of TCP trace records

trigonometry

acos(3F) — Fortran arccosine intrinsic function
asin(3F) — Fortran arcsine intrinsic function
atan2(3F) — Fortran arctangent intrinsic function
atan(3F) — Fortran arctangent intrinsic function
cos(3F) — Fortran cosine intrinsic function
sin(3F) — provide Fortran sine intrinsic functions
tan(3F) — Fortran tangent intrinsic function
trig(3M) — provide trigonometric functions

Trivial File Transfer Protocol

tftp(1C) — transfers files via the Trivial File Transfer Protocol (TFTP)
tftpd(1M) — responds to requests to use the DARPA Trivial File
Transfer Protocol

troff

checknr(1) — checks nroff/troff files
cw(1) — prepare constant-width text for otroff
deroff(1) — removes nroff/troff, tbl, and eqn constructs
diffmk(1) — marks the differences between two files
eqn(1) — format mathematical text for troff
makedev(1) — prepares troff description files
mm(1) — formats documents that contain nroff and mm macro
formatting requests
mmt(1) — typeset documents that contain troff and mm or mv macro-
formatting requests
mvt(1) — typeset documents that contain troff and mm or mv macro-
formatting requests
otroff(1) — formats text for a specific phototypesetter
pic(1) — preprocesses troff files that contain drawings
psdit(1) — converts troff intermediate format to POSTSCRIPT format
psroff(1) — formats a file through troff so it can be printed on a
POSTSCRIPT printer

soelim(1) — eliminates the source commands from `nroff` input
 tbl(1) — table formatter for `nroff` or `troff`
 tc(1) — interprets `troff` output for use at a vintage display device
 troff(1) — formats and typesets files
 eqnchar(5) — special character definitions for `eqn` and `neqn`
 font(5) — description files for device-independent `troff`
 mptx(5) — the macro package for formatting a permuted index
 ms(5) — text formatting macros
 troff(5) — description of `troff` output language

true and false

test(1) — evaluates conditions
 true(1) — provides truth values

truncation

truncate(2) — truncate a file to a specified length

tuning

kconfig(1M) — tunes kernel parameters for work-load optimization

types, data

ftype(3F) — explicit Fortran type conversion
 xdr(3N) — provide library routines for external data representation
 types(5) — primitive system data types

UFS

newfs(1M) — makes a Berkeley 4.2 (UFS) file system
 tunefs(1M) — tunes a Berkeley 4.2 (UFS) file system
 ufs(4) — UFS file-system format

underlining

ul(1) — filters special underlining sequences imbedded in text for use at a display device

UNIX-to-UNIX system communications

uucp(1C) — copies files from one system to another system
 uuencode(1C) — encode and decode a binary file
 uuglist(1C) — displays the service grades that are available on your system
 uulog(1C) — displays information about `uucp` file transfers
 uuname(1C) — displays the names of systems to which `uucp` and `cu` can connect
 uusend(1C) — sends a file to a remote host
 uustat(1C) — controls `uucp` jobs and provides status information
 uuto(1C) — provide an easy interface to the `uucp` command, using the public directories
 uux(1C) — runs a command on a remote system
 Uutry(1M) — contacts a remote system with debugging on
 uucheck(1M) — checks the `uucp` directories and files
 uucico(1M) — transfers files as specified by `uucp` work files
 uucleanup(1M) — removes old files from the `uucp` spool directory

uucpd(1M) — handles the transfer of files by **uucico** over TCP/IP connections
uudemon.admin(1M) — mails current **uucp** work status to the **uucp** administrator
uudemon.cleanup(1M) — cleans up files in the **uucp** spool directory
uudemon.hour(1M) — processes spooled **uucp** requests
uudemon.poll(1M) — sets up polling for selected systems
uusched(1M) — schedules **uucp** file transfers
uuxqt(1M) — handles requests from remote systems to run commands

unmounting file systems

umount(2) — unmount a file system
umount(2) — remove a file system
umount(3) — unmounts a file system

updaters

make(1) — maintains, updates, and regenerates groups of files
rdist(1) — distributes remote files
sync(1) — updates the superblock
touch(1) — updates access and modification times of a file
updater(1) — updates files between two machines
badblk(1M) — sets or updates bad block information
dp(1M) — performs disk partitioning
eu(1M) — updates autorecovery files
eupdate(1M) — updates important files for autorecovery purposes
yppush(1M) — propagates changed Network Information Service (NIS)

maps

sync(2) — update superblock
yppasswd(3N) — updates a user password on the Network Information Service (NIS) master server
bzb(4) — Block Zero Block file format

user accounts

chfn(1) — changes the real-name field of your password file entry for use by **finger**
chsh(1) — changes the default login shell
finger(1) — displays information about the users on a system
fingerd(1M) — handles requests from remote systems for user information from **finger**

user IDs

setuid(2) — set user and group ID
getpw(3C) — gets a name from UID

user interface, choosing

CommandShell(1) — manages command-interpretation windows and moderates access to the A/UX console window
chsh(1) — changes the default login shell
Login(1M) — logs you in to A/UX by using a graphical user interface

user interface, Macintosh

`cmdo(1)` — builds command lines interactively

`macquery(1M)` — posts a Macintosh alert box to query the user

user names

`cuserid(3P)` — gets a character login name of the user

`cuserid(3S)` — gets a character login name of the user

users, general

`finger(1)` — displays information about the users on a system

`groups(1)` — displays group memberships

`id(1)` — displays user and group IDs and names

`last(1)` — displays login and logout times for each user of the system

`logname(1)` — gets the login name

`rusers(1N)` — produces a login list for local machines (RPC version)

`rwho(1N)` — displays a list of the active users from all of the systems on the local network

`su(1)` — substitutes user ID

`talk(1N)` — talks to another user via the terminal

`users(1)` — reports a list of the users who are logged on to the system

`w(1)` — displays a summary of the current system activity

`who(1)` — reports users who are currently logged in to the system

`whoami(1)` — prints effective current user ID

`adduser(1M)` — adds a user account

`fingerd(1M)` — handles requests from remote systems for user information from `finger`

`mkslipuser(1M)` — creates or updates the Compressed Serial Line/Internet Protocol (CSL/IP) database

`rusersd(1M)` — `rusers` invokes a server for users

`rwall(1M)` — writes to all users over a network

`rwalld(1M)` — invokes the network `rwall` server

`talkd(1M)` — invokes the remote user communication server

`wall(1M)` — writes to all users

`whodo(1M)` — informs you of the current system activity

`getuid(2)` — get real and effective user IDs and group IDs

`setreuid(2)` — set real and effective user ID

`setsid(2P)` — create session and set process group ID

`setuid(2)` — set user and group ID

`cuserid(3P)` — gets a character login name of the user

`cuserid(3S)` — gets a character login name of the user

`logname(3X)` — return login name of user

`rnusers(3N)` — return information about users on remote machines

`ttyslot(3C)` — finds the slot in the `utmp` file of the current user

`slip.user(4)` — database of available Compressed Serial Line/Internet Protocol (CSL/IP) connections

UTMP file

- `getut(3C)` — access a utmp file entry
- `ttyslot(3C)` — finds the slot in the utmp file of the current user
- `utmp(4)` — utmp and wtmp entry formats

UUCP

- `uucp(1C)` — copies files from one system to another system
- `uuencode(1C)` — encode and decode a binary file
- `uuglist(1C)` — displays the service grades that are available on your system
- `uulog(1C)` — displays information about uucp file transfers
- `uuname(1C)` — displays the names of systems to which uucp and cu can connect
- `uusend(1C)` — sends a file to a remote host
- `uustat(1C)` — controls uucp jobs and provides status information
- `uuto(1C)` — provide an easy interface to the uucp command, using the public directories
- `uux(1C)` — runs a command on a remote system
- `Uutry(1M)` — contacts a remote system with debugging on
- `uucheck(1M)` — checks the uucp directories and files
- `uucico(1M)` — transfers files as specified by uucp work files
- `uucleanup(1M)` — removes old files from the uucp spool directory
- `uucpd(1M)` — handles the transfer of files by uucico over TCP/IP connections
- `uudemon.admin(1M)` — mails current uucp work status to the uucp administrator
- `uudemon.cleanup(1M)` — cleans up files in the uucp spool directory
- `uudemon.hour(1M)` — processes spooled uucp requests
- `uudemon.poll(1M)` — sets up polling for selected systems
- `uusched(1M)` — schedules uucp file transfers
- `uuxqt(1M)` — handles requests from remote systems to run commands

variables, system

- `kconfig(1M)` — tunes kernel parameters for work-load optimization
- `sysconf(3P)` — gets configurable system variables

version control

- `admin(1)` — creates and administers SCCS files
- `cdc(1)` — changes the delta commentary of an SCCS delta
- `ci(1)` — checks in RCS revisions
- `co(1)` — checks out RCS revisions
- `comb(1)` — combines SCCS deltas
- `delta(1)` — makes a delta (change) to an SCCS file
- `get(1)` — gets a version of an SCCS file
- `help(1)` — provides help information about SCCS commands and messages
- `prs(1)` — displays information about an SCCS file

`racs(1)` — creates new RCS files or changes attributes of existing RCS files
`racsdiff(1)` — compares RCS revisions
`racsintro(1)` — introduces RCS commands
`racsmerge(1)` — merges two versions of an RCS file
`rlog(1)` — displays log messages and other information about RCS files
`rmidel(1)` — removes a delta from an SCCS file
`sact(1)` — displays who has checked a Source Code Control System (SCCS) file out for editing
`sacs(1)` — performs SCCS subsystem commands
`sacsdiff(1)` — compares two versions of an SCCS file
`unget(1)` — undoes a previous get of an SCCS file
`val(1)` — validate SCCS file
`vc(1)` — manipulates version control information inside a data stream
`version(1)` — reports version number of files
`what(1)` — reports identification information for a file
`sccstoracs(1M)` — builds an RCS file from an SCCS file

version control, RCS

`ci(1)` — checks in RCS revisions
`co(1)` — checks out RCS revisions
`racs(1)` — creates new RCS files or changes attributes of existing RCS files
`racsdiff(1)` — compares RCS revisions
`racsintro(1)` — introduces RCS commands
`racsmerge(1)` — merges two versions of an RCS file
`rlog(1)` — displays log messages and other information about RCS files

version control, SCCS

`admin(1)` — creates and administers SCCS files
`cdc(1)` — changes the delta commentary of an SCCS delta
`comb(1)` — combines SCCS deltas
`delta(1)` — makes a delta (change) to an SCCS file
`get(1)` — gets a version of an SCCS file
`help(1)` — provides help information about SCCS commands and messages
`prs(1)` — displays information about an SCCS file
`rmidel(1)` — removes a delta from an SCCS file
`sact(1)` — displays who has checked a Source Code Control System (SCCS) file out for editing
`sacs(1)` — performs SCCS subsystem commands
`sacsdiff(1)` — compares two versions of an SCCS file
`unget(1)` — undoes a previous get of an SCCS file
`val(1)` — validate SCCS file
`vc(1)` — manipulates version control information inside a data stream
`what(1)` — reports identification information for a file
`sccstoracs(1M)` — builds an RCS file from an SCCS file

view graphs

`mvt(1)` — typeset documents that contain `troff` and `mm` or `mv` macro-formatting requests

`mv(5)` — a `troff` macro package for typesetting viewgraphs and slides

windows

`CommandShell(1)` — manages command-interpretation windows and moderates access to the A/UX console window

word breaks

`hyphen(1)` — finds hyphenated words

word counting

`wc(1)` — counts characters, words, and lines in a file

worms

`worm(6)` — plays the game of growing worm

`worms(6)` — plays the game of worms

writing

`write(2)` — write on a file

wumpus

`wump(6)` — plays the game of hunt-the-wumpus

Xerox 1700 terminal

`450(1)` — filters text containing printer control sequences for the DASI terminal

yes (reply to queries)

`yes(1)` — generates `y` entries in response to requests for input

