NAME

ascii − the ASCII character set.

DESCRIPTION

The ASCII character set is as follows:

```
|000 nul|001 soh|002 stx|003 etx|004 eot|005 enq|006 ack|007 bel|
|010 bs |011 ht |012 nl |013 vt |014 np |015 cr |016 so |017 si |
|020 dle|021 dc1|022 dc2|023 dc3|024 dc4|025 nak|026 syn|027 etb|
|030 can|031 em |032 sub|033 esc|034 fs |035 gs |036 rs |037 us |
|040 sp |041  ! |042  " |043  # |044  $ |045  % |046  & |047  ’ |
|050  ( |051  ) |052  * |053  + |054  , |055  - |056  . |057  / |
|060  0 |061  1 |062  2 |063  3 |064  4 |065  5 |066  6 |067  7 |
|070  8 |071  9 |072  : |073  ; |074  < |075  = |076  > |077  ? |
|100  @ |101  A |102  B |103  C |104  D |105  E |106  F |107  G |
|110  H |111  I |112  J |113  K |114  L |115  M |116  N |117  O |
|120  P |121  Q |122  R |123  S |124  T |125  U |126  V |127  W |
|130  X |131  Y |132  Z |133  [ |134  \ |135  ] |136  ˆ |137  _ |
|140  ‘ |141  a |142  b |143  c |144  d |145  e |146  f |147  g |
|150  h |151  i |152  j |153  k |154  l |155  m |156  n |157  o |
|160  p |161  q |162  r |163  s |164  t |165  u |166  v |167  w |
|170  x |171  y |172  z |173  { |174  | |175  } |176  ˜ |177 del|
```

```
|00 nul|01 soh|02 stx|03 etx|04 eot|05 enq|06 ack|07 bel|
|08 bs |09 ht |0A nl |0B vt |0C np |0D cr |0E so |0F si |
|10 dle|11 dc1|12 dc2|13 dc3|14 dc4|15 nak|16 syn|17 etb|
|18 can|19 em |1A sub|1B esc|1C fs |1D gs |1E rs |1F us |
|20 sp |21  ! |22  " |23  # |24  $ |25  % |26  & |27  ’ |
|28  ( |29  ) |2A  * |2B  + |2C  , |2D  - |2E  . |2F  / |
|30  0 |31  1 |32  2 |33  3 |34  4 |35  5 |36  6 |37  7 |
|38  8 |39  9 |3A  : |3B  ; |3C  < |3D  = |3E  > |3F  ? |
|40  @ |41  A |42  B |43  C |44  D |45  E |46  F |47  G |
|48  H |49  I |4A  J |4B  K |4C  L |4D  M |4E  N |4F  O |
|50  P |51  Q |52  R |53  S |54  T |55  U |56  V |57  W |
|58  X |59  Y |5A  Z |5B  [ |5C  \ |5D  ] |5E  ˆ |5F  _ |
|60  ‘ |61  a |62  b |63  c |64  d |65  e |66  f |67  g |
|68  h |69  i |6A  j |6B  k |6C  l |6D  m |6E  n |6F  o |
|70  p |71  q |72  r |73  s |74  t |75  u |76  v |77  w |
|78  x |79  y |7A  z |7B  { |7C  | |7D  } |7E  ˜ |7F del|
```

NAME

environ − user environment

SYNOPSIS

extern char *const *environ;

DESCRIPTION

An array of strings called the ‘environment’ is made available by **execve**(2) when a process begins. By convention these strings have the form ‘*name=value*’. The following names are used by various commands:

PATH The sequence of directory prefixes that **sh**, **time**, **nice**(1), etc., apply in searching for a file known by an incomplete path name. The prefixes are separated by ‘:’. Login shells set **PATH=:/bin:/usr/bin**. Note that the empty space between the ‘=’ and the ‘:’ indicates the current directory. Security aware people move the extra ‘:’ to the end of their path or omit it.

HOME A user’s login directory, set by **login**(1) from the password file **passwd**(5).

TERM The kind of terminal for which output is to be prepared. This information is used by programs that wish to exploit special terminal capabilities, a screen oriented text editor for instance. The terminal type is set by **login**(1) from the tty database **ttytab**(5).

SHELL The file name of the users login shell, set by **login**(1) from the password file **passwd**(5).

TERMCAP
The string describing the terminal in TERM, or the name of the termcap file, see **termcap**(5), **termcap**(3).

EXINIT A startup list of commands read by **elvis**(1).

USER The login name of the user, set by **login**(1) from the password file **passwd**(5).

LOGNAME
Set to the same value as **USER**. BSD derived systems have **USER**, System V has **LOGNAME**, so modern systems have both to avoid problems.

Further names may be placed in the environment by the
export command and ‘*name=value*’ arguments in **sh**(1). Arguments may also be placed in their environment by programs if they use **putenv**(3). Or in the environment of another program by building a new environment for one of the exec functions that accepts an environment list, like **execle**(2) or **execve**(2). It is unwise to conflict with certain **sh**(1) variables that are frequently set and/or exported by ‘.profile’ files: **MAIL**, **PS1**, **PS2**, **IFS**.

SEE ALSO

elvis(1), **login**(1), **sh**(1), **execl**(2), **execve**(2), **system**(3), **termcap**(3), **termcap**(5), **ttytab**(5).

NAME

hier − file system hierarchy

DESCRIPTION

A tour through the Minix directory hierarchy.

/ root

/bin/ Utility programs, cf /usr/bin/ (1)

- cat Show files, **cat**(1)
- sh Shell, **sh**(1)
- mount Mount file systems, **mount**(8)
- ...

/boot Boot Monitor, **monitor**(8)

/dev/ devices (4)

- console
  Main console, **tty**(4)
- tty* Terminals, **tty**(4)
- hd* Hard disk, **hd**(4)
- fd* Floppy disk, **fd**(4)
- ...

/etc/ System configuration and data files, cf /usr/etc/ (1,5,8)

- ethers Ethernet addresses database, **ethers**(5)
- fstab File system table, **fstab**(5)
- group Groups database, **group**(5)
- hostname.file
  Name of the local system, **usage**(5)
- hosts TCP/IP hosts database, **hosts**(5)
- hosts.equiv
  Network equivalent hosts, **hosts.equiv**(5)
- keymap Keymap for custom keyboard, **loadkeys**(8)
- motd Message of the day, **login**(1)
- mtab Mounted file system table, **fstab**(5)
- passwd User database, **passwd**(5)
- psdatabase
  Kernel info for ps command, **ps**(1)
- profile
  System wide shell profile, **sh**(1)
- rc System startup script, **boot**(8)
- resolv.conf
  TCP/IP domain name system, **resolv.conf**(5)
- services
  TCP/IP names to services, **services**(5)
- shadow Shadow password database, **passwd**(5)
- termcap
  Terminal type descriptions, **termcap**(5)
- ttys Active terminals, **init**(8)
- ttytype
  Terminal types, **login**(1)
- utmp User login database, **login**(1)
- ...

/minix
  Minix kernel image, **monitor**(8)

/tmp/ Small, short-lived temporary files, cf /usr/tmp/

/usr/ Home directories and more system files

adm/ System administration files (1,5,8)
- wtmp
  - User logins and logouts, **login**(1)
- ...

ast/ Honorary home directory of Andrew S. Tanenbaum
Also new user initial files

user/ Home directory of *user*
- .ashrc Ash (shell) startup, **ash**(1)
- .ellepro.b1
  - Elle (editor) startup (compiled), **elle**(1)
- .ellepro.e
  - Elle (editor) startup (text), **elle**(1)
- .exrc Ex/vi (editor) startup, **ex**(1)
- .profile
  - Custom user shell profile, **sh**(1)
- .rhosts Remote user permission file, **rhosts**(5)
- ...

bin/ User programs
- cc C compiler, **cc**(1)
- cp Copy files, **cp**(1)
- ls List files, **ls**(1)
- man Show manual pages, **man**(1)
- ...

etc/ More system data files, cf /etc (8)
- ...

include/
C-compiler include files
- minix/ Minix kernel include files
  - ...
- ...

lib/ Compiler libraries and other support stuff
- cawf/ Text formatter support files, **cawf**(1)
- crontab
  - Cron jobs, **cron**(8)
- dict/ Word lists
  - words American English word list, **look**(1)
  - ...
- libc.a C library (Minix-8086 only), **cc**(1)
- *arch* Per architecture compiler binaries and libaries, **cc**(1)
- ...

local/ Local software, cf /usr/
- bin/ Local utilities
- man/ Local manual pages
- src/ Local sources
- ...

man/ Manual pages, **man**(1)
- cat*/ Preformatted manual pages
- man0/ Section 0, Book style user commands
- man1/ User commands
- man2/ System calls
- man3/ Library routines

	man4/	Device files	
	man5/	File formats	
	man6/	Games	
	man7/	Miscellaneous	
	man8/	System utilities	
	whatis	Table of manual pages, **whatis**(5)	
	...		
mdec/			
	boot	Bootstrap code, **installboot**(8)	
	...		
preserve/			
	Saved elvis editor buffers, **elvprsv**(8), **elvrec**(1)		
spool/	Mail and command spooling		
	at/	At jobs, **at**(1)	
		past/	Completed at jobs
	mail/	Mail drops, **mail**(1)	
		user	Mailbox of *user*
		...	
	...		
src/	System and command sources (home of bin)		
	LICENSE		
		Minix license to use	
	commands/		
		Utility sources	
	crclist		
		CRC checksums of the source tree, **srccrc**(8)	
	lib/	Library sources	
	fs/	File system	
	inet/	TCP/IP task	
	kernel/		
		Kernel	
	mm/	Memory manager	
	boot/	Boot Monitor	
	tools/	Kernel image making tools, **tools**(8)	
tmp/	Large temporary files		

SEE ALSO

ls(1), **man**(1), **find**(1), **grep**(1), **checkhier**(8).

NOTES

Not all of the directories and files shown are present. They must be created as needed.

BUGS

Many of the listed manual references do not yet exist.

AUTHOR

Kees J. Bot (kjb@cs.vu.nl)

NAME

man - nroff macro package for manual pages

SYNOPSIS

nroff −man *file* ...

DESCRIPTION

These macros are used to lay out reference pages for manuals.

Any text argument *t* may be zero to six words. Quotes may be used to include blanks in a 'word'. *Text* can be empty, but unlike normal UNIX macros, the next line is not used.

A prevailing indent distance is remembered between successive indented paragraphs, and is reset to default value upon reaching a non-indented paragraph (i.e. at .SH or .SS).

FILES

/usr/lib/tmac/tmac.an For standard Minix nroff.

/usr/lib/cawf/man.mac For cawf.

SEE ALSO

nroff(1), **man**(1).

REQUEST SUMMARY

Request	Cause Break?	Explanation
.B t	no	Text t is bold. Quote to imbed blanks.
.I t	no	Text t is italic. Quote to imbed blanks.
.IP x	yes	Set prevailing indent to 5. Begin indented paragraph with hanging tag given by first argument.Tag x is always placed on a separate line.
.LP	yes	Same as .PP.
.PP	yes	Begin paragraph.Set prevailing indent to 5.
.RE	yes	End of relative indent.Set prevailing indent to amount of starting .RS.
.RS	yes	Start relative indent, move left margin in distance 5.
.SH t	yes	Subhead. Quote to imbed blanks.
.SS t	yes	Subsection. Quote to imbed blanks. No indent for t.
.TH n s c v d	yes	Begin page named n of chapter s; c is the chapter name; d is the date of the most recent change; v is version number. Sets prevailing indent and tabs to 5.

EXAMPLE

The following illustrates some of the requests available with this macro package:

```
.\" this is a comment
.TH DEMO 1
.SH NAME
demo \- show how to use \-man package
.SH SYNOPSIS
.B demo
.RI [ options ]
.IR file " ..."
.SH DESCRIPTION
This is a test for showing how to use the
.BR nroff (1)
```

```
man package. It shows how to use .TH, .SH, .PP, .B, .I, and .IP
commands.
.PP
This will be a new paragraph. You can also use normal
.BR nroff (1)
commands in the text.
.SS Nroff Commands
.IP '\e"'
This is the comment command.  \" You won't see this.
.IP nf
No fill mode (the normal mode is fill mode where things
get justified right and left).
.IP fi
Re-enter fill mode.
.IP br
Break line here no matter what.
.IP sp
Vertical space (also causes a break to occur).
.sp
Note that to continue an indent and make a new paragraph (as
is the case here), just put in a space (.sp).
.PP
Now we should be at a new paragraph.
```

Executing **nroff −man demo.man** results in the following output: (Ignoring page headers and footers)

NAME

demo \- show how to use \-man package

SYNOPSIS

demo [*options*] *file* ...

DESCRIPTION

This is a test for showing how to use the **nroff**(1) man package. It shows how to use .TH, .SH, .PP, .B, .I, and .IP commands.

This will be a new paragraph. You can also use normal **nroff**(1) commands in the text.

Nroff Commands

'\"' This is the comment command.

nf No fill mode (the normal mode is fill mode where things get justified right and left).

fi Re-enter fill mode.

br Break line here no matter what.

sp Vertical space (also causes a break to occur).

Note that to continue an indent and make a new paragraph (as is the case here), just put in a space (.sp).

Now we should be at a new paragraph.

CONVENTIONS

A typical manual page for a command or function is laid out as follows:

```
.TH TITLE [1-8]
    The name of the command or function in upper-case,
    which serves as the title of the manual page. This is
```

followed by the number of the section in which it appears.

.SH NAME

name - one-line summary

The name, or list of names, by which the command is called, followed by a dash and then a one-line summary of the action performed. All in roman font, this section contains no troff(1) commands or escapes, and no macro requests. It is used to generate the whatis(1) database.

.SH SYNOPSIS

Commands:

The syntax of the command and its arguments as typed on the command line. When in boldface, a word must be typed exactly as printed. When in italics, a word can be replaced with text that you supply. Syntactic symbols appear in roman face:

[ ] An argument, when surrounded by brackets is optional.

| Arguments separated by a vertical bar are exclusive. You can supply only item from such a list.

... Arguments followed by an elipsis can be repeated. When an elipsis follows a bracketed set, the expression within the brackets can be repeated.

Functions:

If required, the data declaration, or #include directive, is shown first, followed by the function declaration. Otherwise, the function declaration is shown.

.SH DESCRIPTION

A narrative description of the command or function in detail, including how it interacts with files or data, and how it handles the standard input, standard output and standard error.

Filenames, and references to commands or functions described elswhere in the manual, are italicised. The names of options, variables and other literal terms are in boldface.

.SH OPTIONS

The list of options along with a description of how each affects the commands operation.

.SH ENVIRONMENT
Environment variables used.

.SH FILES
A list of files associated with the command or function.

.SH "SEE ALSO"
A comma-separated list of related manual pages, followed by references to other published materials. This section contains no troff(1) escapes or commands, and no macro requests.

.SH DIAGNOSTICS
A list of diagnostic messages and an explanation of each.

.SH NOTES
Any additional notes such as installation-dependent functionality.

.SH BUGS
A description of limitations, known defects, and possible problems associated with the command or function.

.SH AUTHOR
The program's author and any pertinent release info.

.SH VERSION
The program's current version number and release date.

BUGS

Even though **cawf**(1) has a better chance at formatting a random manual page then the standard Minix nroff, it has two annoying bugs in its macro set. Both .PP and .IP reset the indentation level to the level set by .SH. This means that you can't use them in a piece of text indented by .RS. For .IP this is troublesome, you can see why in the unformatted source of this text. .PP can simply be replaced by .sp, or better yet, by .SP with the following macro defined somewhere in your text:

```
.de SP
.if t .sp 0.4
.if n .sp
..
```

This will make .SP use 4/10 of a line if formatted by troff, just like .PP.