NAME

groff_char − groff character names

DESCRIPTION

This manual page lists the standard **groff** input characters. The output characters in this document will look different depending on which output device was chosen (with option **−T** for the **man**(1) program or the roff formatter). Only the characters that are available for the device that is being used to print or view this manual page will be displayed (the device currently used is 'ps').

In the actual version, **groff** provides only 8-bit characters for direct input and named characters for further glyphs. On ASCII platforms, character codes in the range 0 to 127 (decimal) represent the usual 7-bit ASCII characters, while codes between 127 and 255 are interpreted as the corresponding characters in the *Latin-1* (*ISO-8859-1*) code set. On EBCDIC platforms, only the code page **cp1047** is supported (which contains the same characters as Latin-1). It is rather straightforward (for the experienced user) to set up other 8bit encodings like *Latin-2*; since **groff** will use Unicode in the next major version, no additional encodings are provided.

All roff systems provide the concept of named characters. In traditional roff systems, only names of length 2 were used, while groff also provides support for longer names. It is strongly suggested that only named characters are used for all characters outside of the 7-bit ASCII range.

Some of the predefined groff escape sequences (with names of length 1) also produce single characters; these exist for historical reasons or are printable versions of syntactical characters. They include \\, \', \`, \-, \., and **e**; see **groff**(7).

In groff, all of these different types of characters can be tested positively with the **.if c** conditional.

REFERENCE

In this section, the characters in groff are specified in tabular form. The meaning of the columns is as follows.

Output shows how the character is printed for the current device; although this can have quite a different shape on other devices, it always represents the same glyph.

Input name
specifies how the character is input either directly by a key on the keyboard, or by a groff escape sequence.

Input code
applies to characters which can be input with a single character, and gives the ISO Latin-1 decimal code of that input character. Note that this code is equivalent to the lowest 256 Unicode characters; (including 7-bit ASCII in the range 0 to 127).

PostScript name
gives the usual PostScript name of the output character.

ASCII Characters

These are the basic characters having 7-bit ASCII code values. These are identical to the first 127 characters of the character standards ISO-8859-1 (Latin-1) and Unicode (range *C0 Controls and Basic Latin*). To save space, not every code has an entry in the following because the following code ranges are well known.

0−32 Control characters (print as themselves).

48−57 Decimal digits 0 to 9 (print as themselves).

65−90 Upper case letters A−Z (print as themselves).

97−122 Lower case letters a−z (print as themselves).

127 Control character (prints as itself).

The remaining ranges constitute the printable, non-alphanumeric ASCII characters; only these are listed below. As can be seen in the table below, most of these characters print as themselves; the only exceptions are the following characters:

`` ` `` the ISO Latin-1 ‘Grave Accent’ (code 96) prints as ‘, a left single quotation mark,

`' ` the ISO Latin-1 ‘Apostrophe’ (code 39) prints as ’, a right single quotation mark; the corresponding ISO Latin-1 characters can be obtained with **\‘** and **\(aq**.

`-` the ISO Latin-1 ‘Hyphen, Minus Sign’ (code 45) prints as a hyphen; a minus sign can be obtained with **\-**.

`~` the ISO Latin-1 ‘Tilde’ (code 126); a larger glyph can be obtained with **\(ti**.

`^` the ISO Latin-1 ‘Circumflex Accent’ (code 94); a larger glyph can be obtained with **\(ha**.

Output	*Input name*	*Input code*	*PostScript name*	*Notes*
!	!	33	exclam	
"	"	34	quotedbl	
#	#	35	numbersign	
$	$	36	dollar	
%	%	37	percent	
&	&	38	ampersand	
’	'	39	quoteright	
(	(	40	parenleft	
)	)	41	parenright	
*	*	42	asterisk	
+	+	43	plus	
,	,	44	comma	
-	-	45	hyphen	
.	.	46	period	
/	/	47	slash	
:	:	58	colon	
;	;	59	semicolon	
<	<	60	less	
=	=	61	equal	
>	>	62	greater	
?	?	63	question	
@	@	64	at	
[	[	91	bracketleft	
\	\	92	backslash	
]	]	93	bracketright	
^	^	94	circumflex	circumflex accent
_	_	95	underscore	
‘	`	96	quoteleft	
{	{	123	braceleft	
|	|	124	bar	
}	}	125	braceright	
~	~	126	tilde	tilde accent

Latin-1 Special Characters

These characters have character codes between 128 and 255. They are interpreted as characters according to the *Latin-1* (*iso-8859-1*) code set, being identical to the Unicode range *C1 Controls and Latin-1 Supplement*.

128−159
: the C1 Controls; they print as themselves, but the effect is mostly undefined.

160 the ISO Latin-1 *no-break space* is mapped to **‘\ ’**, the escaped space character.

173 the soft hyphen control character (prints as itself). groff never use this character for output (thus it is omitted in the table below); the input character 173 is mapped onto **\%**.

The remaining ranges (161−172, 174−255), called the *Latin-1 Supplement* in Unicode, are printable characters that print as themselves. Although they can be specified directly with the keyboard on systems with a Latin-1 code page, it is better to use their named character equivalent; see next section.

Output	*Input name*	*Input code*	*PostScript name*	*Notes*
¡	¡	161	exclamdown	inverted exclamation mark
¢	¢	162	cent	
£	£	163	sterling	
¤	¤	164	currency	
¥	¥	165	yen	
¦	¦	166	brokenbar	
§	§	167	section	
¨	¨	168	dieresis	
©	©	169	copyright	
ª	ª	170	ordfeminine	
«	«	171	guillemotleft	
¬	¬	172	logicalnot	
®	®	174	registered	
¯	¯	175	macron	
°	°	176	degree	
±	±	177	plusminus	
²	²	178	twosuperior	
³	³	179	threesuperior	
´	´	180	acute	acute accent
µ	µ	181	mu	micro sign
¶	¶	182	paragraph	
·	·	183	periodcentered	
¸	¸	184	cedilla	
¹	¹	185	onesuperior	
º	º	186	ordmasculine	
»	»	187	guillemotright	
¼	¼	188	onequarter	
½	½	189	onehalf	
¾	¾	190	threequarters	
¿	¿	191	questiondown	
À	À	192	Agrave	
Á	Á	193	Aacute	
Â	Â	194	Acircumflex	
Ã	Ã	195	Atilde	
Ä	Ä	196	Adieresis	
Å	Å	197	Aring	
Æ	Æ	198	AE	
Ç	Ç	199	Ccedilla	
È	È	200	Egrave	
É	É	201	Eacute	
Ê	Ê	202	Ecircumflex	
Ë	Ë	203	Edieresis	
Ì	Ì	204	Igrave	
Í	Í	205	Iacute	
Î	Î	206	Icircumflex	
Ï	Ï	207	Idieresis	
Ð	Ð	208	Eth	
Ñ	Ñ	209	Ntilde	

Output	*Input name*	*Input code*	*PostScript name*	*Notes*
Ò	Ò	210	Ograve	
Ó	Ó	211	Oacute	
Ô	Ô	212	Ocircumflex	
Õ	Õ	213	Otilde	
Ö	Ö	214	Odieresis	
×	×	215	multiply	
Ø	Ø	216	Oslash	
Ù	Ù	217	Ugrave	
Ú	Ú	218	Uacute	
Û	Û	219	Ucircumflex	
Ü	Ü	220	Udieresis	
Ý	Ý	221	Yacute	
Þ	Þ	222	Thorn	
ß	ß	223	germandbls	
à	à	224	agrave	
á	á	225	aacute	
â	â	226	acircumflex	
ã	ã	227	atilde	
ä	ä	228	adieresis	
å	å	229	aring	
æ	æ	230	ae	
ç	ç	231	ccedilla	
è	è	232	egrave	
é	é	233	eacute	
ê	ê	234	ecircumflex	
ë	ë	235	edieresis	
ì	ì	236	igrave	
í	í	237	iacute	
î	î	238	icircumflex	
ï	ï	239	idieresis	
ð	ð	240	eth	
ñ	ñ	241	ntilde	
ò	ò	242	ograve	
ó	ó	243	oacute	
ô	ô	244	ocircumflex	
õ	õ	245	otilde	
ö	ö	246	odieresis	
÷	÷	247	divide	
ø	ø	248	oslash	
ù	ù	249	ugrave	
ú	ú	250	uacute	
û	û	251	ucircumflex	
ü	ü	252	udieresis	
ý	ý	253	yacute	
þ	þ	254	thorn	
ÿ	ÿ	255	ydieresis	

Named Characters

The named character idiom is the standard way to specify special characters in roff systems. They can be embedded into the document text by using escape sequences. **groff**(7) describes how these escape sequences look. The character names can consist of quite arbitrary characters from the ASCII or Latin-1 code set, not only alphanumeric characters. Here some examples:

c named character having the name *c*, which consists of a single character (length 1).

\(*ch* named character having the 2-character name *ch*.

\[*char_name***]**
named character having the name *char_name* (having length 1, 2, 3, . . .).

In groff, each 8bit input character can also referred to by the construct **\n[char***n***]** where *n* is the decimal code of the character, a number between 0 and 255 without leading zeros. They are mapped onto glyph entities using the **.trin** request. Moreover, new character names can be created by the **.char** request; see **groff**(7).

Output	*Input name*	*PostScript name*	*Notes*
Ð	\[-D]	Eth	Icelandic uppercase eth
ð	\[Sd]	eth	Icelandic lowercase eth
Þ	\[TP]	Thorn	Icelandic uppercase thorn
þ	\[Tp]	thorn	Icelandic lowercase thorn
ß	\[ss]	germandbls	German sharp s
Ligatures			
ff	\[ff]	ff	ff ligature
fi	\[fi]	fi	fi ligature
fl	\[fl]	fl	fl ligature
ffi	\[Fi]	ffi	ffi ligature
ffl	\[Fl]	ffl	ffl ligature
Æ	\[AE]	AE	
æ	\[ae]	ae	
Œ	\[OE]	OE	
œ	\[oe]	oe	
IJ	\[IJ]	IJ	Dutch IJ ligature
ij	\[ij]	ij	Dutch ij ligature
ı	\[.i]	dotlessi	i without a dot (Turkish)
Accented Characters			
Á	\['A]	Aacute	
É	\['E]	Eacute	
Í	\['I]	Iacute	
Ó	\['O]	Oacute	
Ú	\['U]	Uacute	
Ý	\['Y]	Yacute	
á	\['a]	aacute	
é	\['e]	eacute	
í	\['i]	iacute	
ó	\['o]	oacute	
ú	\['u]	uacute	
ý	\['y]	yacute	
Ä	\[:A]	Adieresis	A with umlaut
Ë	\[:E]	Edieresis	
Ï	\[:I]	Idieresis	
Ö	\[:O]	Odieresis	
Ü	\[:U]	Udieresis	
Ÿ	\[:Y]	Ydieresis	
ä	\[:a]	adieresis	
ë	\[:e]	edieresis	
ï	\[:i]	idieresis	
ö	\[:o]	odieresis	

Output	*Input name*	*PostScript name*	*Notes*
ü	\[:u]	udieresis	
ÿ	\[:y]	ydieresis	
Â	\[^A]	Acircumflex	
Ê	\[^E]	Ecircumflex	
Î	\[^I]	Icircumflex	
Ô	\[^O]	Ocircumflex	
Û	\[^U]	Ucircumflex	
â	\[^a]	acircumflex	
ê	\[^e]	ecircumflex	
î	\[^i]	icircumflex	
ô	\[^o]	ocircumflex	
û	\[^u]	ucircumflex	
À	\[`A]	Agrave	
È	\[`E]	Egrave	
Ì	\[`I]	Igrave	
Ò	\[`O]	Ograve	
Ù	\[`U]	Ugrave	
à	\[`a]	agrave	
è	\[`e]	egrave	
ì	\[`i]	igrave	
ò	\[`o]	ograve	
ù	\[`u]	ugrave	
Ã	\[~A]	Atilde	
Ñ	\[~N]	Ntilde	
Õ	\[~O]	Otilde	
ã	\[~a]	atilde	
ñ	\[~n]	ntilde	
õ	\[~o]	otilde	
Š	\[vS]	Scaron	
š	\[vs]	scaron	
Ž	\[vZ]	Zcaron	
ž	\[vz]	zcaron	
Ç	\[,C]	Ccedilla	
ç	\[,c]	ccedilla	
Ł	\[/L]	Lslash	Polish L with a slash
ł	\[/l]	lslash	Polish l with a slash
Ø	\[/O]	Oslash	Scandinavic slashed O
ø	\[/o]	oslash	Scandinavic slashed o
Å	\[oA]	Aring	
å	\[oa]	aring	
Accents			
˝	\[a"]	hungarumlaut	Hungarian umlaut
¯	\[a-]	macron	macron or bar accent
˙	\[a.]	dotaccent	dot accent
^	\[a^]	circumflex	circumflex accent
´	\[aa]	acute	acute accent
`	\[ga]	grave	grave accent
˘	\[ab]	breve	breve accent
¸	\[ac]	cedilla	cedilla accent
¨	\[ad]	dieresis	umlaut or dieresis
ˇ	\[ah]	caron	háček accent
°	\[ao]	ring	ring or circle accent

Output	*Input name*	*PostScript name*	*Notes*
˜	\[a~]	tilde	tilde accent
˛	\[ho]	ogonek	hook or ogonek accent
^	\[ha]	asciicircum	ASCII circumflex, hat, caret
~	\[ti]	asciitilde	ASCII tilde, large tilde
Quotes			
„	\[Bq]	quotedblbase	low double comma quote
‚	\[bq]	quotesinglbase	low single comma quote
“	\[lq]	quotedblleft	
”	\[rq]	quotedblright	
‘	\[oq]	quoteleft	single open quote
’	\[cq]	quoteright	single closing quote (ASCII 39)
'	\[aq]	quotesingle	apostrophe quote
"	\[dq]	quotedbl	double quote (ASCII 34)
«	\[Fo]	guillemotleft	
»	\[Fc]	guillemotright	
‹	\[fo]	guilsinglleft	
›	\[fc]	guilsinglright	
Punctuation			
¡	\[r!]	exclamdown	
¿	\[r?]	questiondown	
—	\[em]	emdash	em dash
–	\[en]	endash	en dash
-	\[hy]	hyphen	
Brackets			
[	\[lB]	bracketleft	
]	\[rB]	bracketright	
{	\[lC]	braceleft	
}	\[rC]	braceright	
⟨	\[la]	angleleft	left angle bracket
⟩	\[ra]	angleright	right angle bracket
Arrows			
←	\[<-]	arrowleft	
→	\[->]	arrowright	
↔	\[<>]	arrowboth	horizontal double-headed arrow
↓	\[da]	arrowdown	
⇑	\[ua]	arrowup	
⇕	\[va]	arrowupdn	vertical double-headed arrow
⇐	\[lA]	arrowdblleft	
⇒	\[rA]	arrowdblright	
⇔	\[hA]	arrowdblboth	horizontal double-headed double arrow
⇓	\[dA]	arrowdbldown	
⇑	\[uA]	arrowdblup	
—	\[an]	arrowhorizex	horizontal arrow extension
Lines			
|	\[or]	bar	
|	\[ba]	bar	
|	\[br]	br	box rule with traditional troff metrics
_	\[ru]	ru	baseline rule
	\[ul]	ul	underline with traditional troff metrics
⏐	\[bv]	bv	bar vertical

Output	*Input name*	*PostScript name*	*Notes*
¦	\[bb]	brokenbar	
/	\[sl]	slash	
\	\[rs]	backslash	
Text markers			
○	\[ci]	circle	
•	\[bu]	bullet	
‡	\[dd]	daggerdbl	double dagger sign
†	\[dg]	dagger	
◊	\[lz]	lozenge	
□	\[sq]	square	
¶	\[ps]	paragraph	
§	\[sc]	section	
☜	\[lh]	handleft	
☞	\[rh]	handright	
@	\[at]	at	
#	\[sh]	numbersign	
↵	\[CR]	carriagereturn	carriage return symbol
✓	\[OK]	a19	check mark, tick
Legalize			
©	\[co]	copyright	
®	\[rg]	registered	
™	\[tm]	trademark	
Currency symbols			
$	\[Do]	dollar	
¢	\[ct]	cent	
¥	\[Ye]	yen	
£	\[Po]	sterling	British currency sign
¤	\[Cs]	currency	Scandinavian currency sign
f	\[Fn]	florin	Dutch currency sign
Units			
°	\[de]	degree	
‰	\[%0]	perthousand	per thousand, per mille sign
′	\[fm]	minute	footmark, prime
″	\[sd]	second	
µ	\[mc]	mu	micro sign
ª	\[Of]	ordfeminine	
º	\[Om]	ordmasculine	
Logical Symbols			
∧	\[AN]	logicaland	
∨	\[OR]	logicalor	
¬	\[no]	logicalnot	
∃	\[te]	existential	there exists, existential quantifier
∀	\[fa]	universal	for all, universal quantifier
∋	\[st]	suchthat	
∴	\[3d]	therefore	
∴	\[tf]	therefore	
Mathematical Symbols			
½	\[12]	onehalf	
¼	\[14]	onequarter	
¾	\[34]	threequarters	

Output	*Input name*	*PostScript name*	*Notes*
¹	\[S1]	onesuperior	
²	\[S2]	twosuperior	
³	\[S3]	threesuperior	
+	\[pl]	plusmath	plus sign in special font
−	\-	minus	minus sign from current font
±	\[+-]	plusminus	
±	\[t+-]		plusminustext variant of '+-'
·	\[pc]	periodcentered	multiplication dot
·	\[md]	dotmath	
×	\[mu]	multiply	
×	\[tmu]		multiplytext variant of 'mu'
⊗	\[c*]	circlemultiply	multiply sign in a circle
⊕	\[c+]	circleplus	plus sign in a circle
÷	\[di]	divide	division sign
÷	\[tdi]		dividetext variant of 'di'
⁄	\[f/]	fraction	bar for fractions
*	\[**]	asteriskmath	
≤	\[<=]	lessequal	
≥	\[>=]	greaterequal	
≠	\[!=]	notequal	
=	\[eq]	equalmath	equals sign in special font
≡	\[==]	equivalence	
≅	\[=~]	congruent	
~	\[ap]	similar	
≈	\[~~]	approxequal	
≈	\[~=]	approxequal	
∝	\[pt]	proportional	
∅	\[es]	emptyset	
∈	\[mo]	element	
∉	\[nm]	notelement	
⊄	\[nb]	notsubset	
⊂	\[sb]	propersubset	
⊃	\[sp]	propersuperset	
⊆	\[ib]	reflexsubset	
⊇	\[ip]	reflexsuperset	
∩	\[ca]	intersection	intersection, cap
∪	\[cu]	union	union, cup
∠	\[/_]	angle	
⊥	\[pp]	perpendicular	
∫	\[is]	integral	
∑	\[sum]		sum
∏	\[product]		product
∇	\[gr]	gradient	
√	\[sr]	radical	square root
‾	\[rn]		overline
∞	\[if]	infinity	
ℵ	\[Ah]	aleph	
ℑ	\[Im]	Ifraktur	Gothic I, imaginary
ℜ	\[Re]	Rfraktur	Gothic R, real
℘	\[wp]	weierstrass	Weierstrass p
∂	\[pd]	partialdiff	partial differentiation sign

Greek characters

Output	*Input name*	*PostScript name*	*Notes*
A	\[*A]	Alpha	
B	\[*B]	Beta	
Ξ	\[*C]	Xi	
Δ	\[*D]	Delta	
E	\[*E]	Epsilon	
Φ	\[*F]	Phi	
Γ	\[*G]	Gamma	
Θ	\[*H]	Theta	
I	\[*I]	Iota	
K	\[*K]	Kappa	
Λ	\[*L]	Lambda	
M	\[*M]	Mu	
N	\[*N]	Nu	
O	\[*O]	Omicron	
Π	\[*P]	Pi	
Ψ	\[*Q]	Psi	
P	\[*R]	Rho	
Σ	\[*S]	Sigma	
T	\[*T]	Tau	
ϒ	\[*U]	Upsilon	
Ω	\[*W]	Omega	
X	\[*X]	Chi	
H	\[*Y]	Eta	
Z	\[*Z]	Zeta	
α	\[*a]	alpha	
β	\[*b]	beta	
ξ	\[*c]	xi	
δ	\[*d]	delta	
ε	\[*e]	epsilon	
ϕ	\[*f]	phi	
φ	\[+f]	phi1	variant phi
γ	\[*g]	gamma	
θ	\[*h]	theta	
ϑ	\[+h]	theta1	variant theta
ι	\[*i]	iota	
κ	\[*k]	kappa	
λ	\[*l]	lambda	
μ	\[*m]	mu	
ν	\[*n]	nu	
o	\[*o]	omicron	
π	\[*p]	pi	
ϖ	\[+p]	omega1	variant pi, looking like omega
ψ	\[*q]	psi	
ρ	\[*r]	rho	
σ	\[*s]	sigma	
τ	\[*t]	tau	
υ	\[*u]	upsilon	
ω	\[*w]	omega	
χ	\[*x]	chi	
η	\[*y]	eta	
ζ	\[*z]	zeta	
ς	\[ts]	sigma1	terminal sigma

Output	*Input name*	*PostScript name*	*Notes*
Card symbols			
♣	\[CL]	club	club suit
♠	\[SP]	spade	spade suit
♥	\[HE]	heart	heart suit
♦	\[DI]	diamond	diamond suit

AUTHOR

This document is part of *groff*, the GNU roff distribution. It was written by James Clark ⟨jjc@jclark.com⟩ with additions by Werner Lemberg ⟨wl@gnu.org⟩ and Bernd Warken ⟨bwarken@mayn.de⟩.

SEE ALSO

groff(1)
: the GNU roff formatter.

groff(7)
: a short reference of the groff formatting language.

An extension to the troff character set for Europe, E.G. Keizer, K.J. Simonsen, J. Akkerhuis; EUUG Newsletter, Volume 9, No. 2, Summer 1989

The Unicode Standard ⟨http://www.unicode.org⟩