Quick Reference Guide
for **mm** and **gtroff** (ver. 3)

All commands are **mm** unless marked **t**. See 'man groff_mm' and 'man 7 groff' for details.

Page Control

Headers are controlled by \nN:

- 0: print on all pages
- 1: page 1 header moved to footer
- 2: no page 1 header
- 3: page numbers as footer (*section no-page no*) and level 1 headings start on a new page
- 4: no header on page 1 and on other pages only print explicit .PH header
- 5: same as 3 and .FG numbering enabled.

Note that \ sequences in header strings need doubling.

.PGFORM [L [P [O [1]]]]
: Sets linelength, pagelength and/or pageoffset. 1 suppresses line break.

.VM [top [bottom]]
: Increase vertical margin [5 0] .VM alone resets.

.SK [pages]
: skip pages

.ne N need vertical space (conditional page feed) **t**

.pn expr N
: set next page number **t**

.OH/.EH/.PH "'L'C'R'"
: odd/even/both page header.

.OF/.EF/.PF "'L'C'R'"
: odd/even/both page footer.

.TP, .PX
: User defined macros instead of header (.TP) or as well as header (.PX)

.PGNH
: suppress header on next page

.OP skip to next odd page

.2C 2 column mode

.1C [1] 1 column mode; 1=no page feed

.MC width [spacing]
: Multi-column mode

.NCOL
: New column

.WC *args*
: Footnote/display style with **.2C** [-WF -FF -WD FB]

Paragraph Control

.P [type]
: New paragraph. 0=left justified. 1=indented. 2=indented except after .H, .DE or .LE

.H level [heading-text [heading-suffix]]
: Numbered heading.

.HU 'title'
: Unnumbered heading.

.nr **Hb** N
: If level ≤ N [2] do .br before text

.nr **Hs** {0-7}
: ½ space after heading level ≤ N [2].

.ds **HP** 0 0 0 0 0 0 0
: Header point size (0=10pt)

.ds **HF** 2 2 2 2 2 2 2
: Fontlist for headings

.nr **Hi** {0-2}
: Post-heading indent [1]. 0=text left justified. 1=use current paragraph style. 2=Indent text to first word of heading.

.nr **Hc** N
: Centre headings at level 1 to N [0].

.nr **Ej** N
: Level ≤ N start on new page.

.nr **Ht** {0-1}
: Suppress subsection number concatenation (4 instead of 3.2.4) [0]

.HM *level-1-style level-2-style ...*
: Heading numbering style with *level-n-style* from:

level-i	**heading number**
1	1, 2, 3, ...
001	001, 002, 003, ...
a	a, b, ... z, aa, bb, ...
A	A, B, ... Z, AA, BB, ...
i	i, ii, iii, ...
I	I, II, III, ...

.FG [*caption* [*modifier* [*control*]]]
: Label a figure. *control* [0] 0:*modifier* is a prefix, 1:*modifier* is a suffix, 2:*modifier* replaces number

.TB [*caption* [*modifier* [*control*]]]
: Label a table (params as .FG).

.EC [*caption* [*modifier* [*control*]]]
: Label an equation (params as .FG).

.TC *l1 n l2 leader-char a1 ... a3*
: Produce table of contents [1 1 2 0].

Lists

In general, the optional parameter [1] suppresses the space after the item.

.LB text-indent mark-indent pad type [mark [LI-space [LB-space]]]
: List begin macro (called by .VL, .AL etc)

.BL [text-indent [1]]
: Start bullet list

.LI [mark [1]]
: List item

.ML mark [text-indent]
: Marked list start

.DL [text-indent [1]]
: Dashed list.

.VL text-indent [mark-indent [1]]
: Variable-item list start

.AL [type [text-indent [1]]]]
: Start autoincrement list

.LE list end

Blocks & displays, footnotes

.DF [format [fill [rindent]]]
: Begin floating display

.DS [format [fill [rindent]]]
: Begin static display [L N 0]. Format: L=left; C=centred; I=indent by \n(Si; CB=centre as block. Fill: N=no fill; F=fill.

.DE End display

.FD [arg [1]]
: footnote default format

.FS [mark] ... **.FE**
: footnote

***F** Insert automatic footnote number

Line Control

.br start new line **t**

.fi fill on **t**

.nf fill off **t**

.na right justification off **t**

.ad b right justification on **t**

.ad l flush left, ragged right **t**

.ad r flush right, ragged left **t**

.ad c centred **t**

.ad previous adjustment mode **t**

.ce n centre 'n' lines (n=0 stops centering) [1] **t**

.\" comment at start of line **t**

\" comment in the line!! **t**

.hy hyphenation on **t**

.nh hyphenation off **t**

.ls 2 double space **t**

.SP [lines]
: vertical space

Character Control

.S [size [spacing]]
: font size. C=current, D=default, P=previous, +/-.

.SM string1 [string2 [string3]]
: small font

.SM string1 [string2 [string3]]
: small font

NB the remainder are **troff**

\& nop - use at the start of line before . or '

\w*word*
: Width of word

\N'xxx'
: octal character e.g. xxx=163:£ (Pound)

\c continue word (if in fill mode)

\c break word (if in nofill mode)

\f[*font*] change to *font*

Font variants: add R, I, B or BI:

A AvantGarde-Book
BM Bookman-Light
C Courier
H Helvetica
HN Helvetica-Narrow
N NewCenturySchoolBook
P Palatino
T TimesRoman

No variants:
ZCMI *ZapfChanceryMediumItalic*
P Previous font
S Symbol
SS Slanted Symbol
ZD ZapfDingBats
ZDR ZapfDingBats Reversed

.sz [±N]
font size - temporary - until macro package resets it! may not play nicely with macro packages.
\s(nn change to (arbitrary) size \s(72
\sn or **\s**nn
change to troff constrained size \s6 \s10 \s36 (max)
\s0 change to previous size
***(dw** Monday etc
***(mo** January
***(td** date: June 1, 1992
\n(dy day of the month
\n(mo month number
\n[year]
Y2K year e.g. "1992". Thus `\n(dy/\n(mo/\n[year]` produces 18/6/2018

Graphics & Motion (all)

Note: | makes coordinates absolute rather than the usual relative.

.hl horizontal line
\D'l x y'
Draw a line to (x, y)
\D'c D'
Draw circle radius D with left side at current position
\D'e D_1 D_1'
draw ellipse of diameters D_1 and D_2
\D'a H_1 V_1 H_2 V_2'
Draw an arc from current position to (H_1+H_2, V_1+V_2)
\D'~ H_1 V_1 H_2 V_2 ...'
Draw a spline from current position using the given tie points.
\h'x' move horizontally to the right
\v'y' move vertically down
\l'x' draw horizontal line
\L'y' draw vertical line
.mk *x*, **\k***x*
Mark present vertical/horizontal position in *x*
.sp |\n*x*u **\h**'|\n*x*u'
Move back to vertical/horizontal mark *x*

TROFF (all)

groff has long names: \n**[name]**. *troff* needs 1- or 2-letter names: \n**a** or \n(**aa**.

.nr pi *n*
set number register pi
\n[*name*]
dereference register *name*
.ds *xx* set string
***[*xx*]** deference string *xx*
\(dg dagger => †
\(dd double dagger => ‡
\(sc section => §
\(rh right hand => ☞
\(lh left hand => ☜
\(bu bullet => •
\(sq square => □
\(14 quarter => ¼
\(12 half => ½
\(34 three-quarters = > ¾
\(de degree => °
\(rg registered => ®
\(co copyright => ©
\(eu Euro => €
\(Po Pound => £
others See `'man groff_char'`
\e backslash (actually, the current escape character)
.so file include *file*
.?? 'message'
puts *message* on stderr
.tm 'message'
puts *message* on stderr
.di 'message'
puts *message* on stderr
.ab 'message'
puts *message* on stderr and aborts
.PSPIC [-L|-R|-I n] file [width [height]]
Draw PostScript (under -Tps only). The graphic may be left or right justified (default centred) or indented by n.

Macros (all)

.de xx start macro xx
.. end macro

Grouping lines:
.if n \{
...
.\}
Beware of characters after **\}**! Use **.\}** at the start of a line.
Forms of **.if**:
.if 'string1'string2'
True if strings equal
.if !'string1'string2'
True if strings not equal
.if n|t|o|e
True if nroff| troff| even page| odd page
.if \nX>0, **.if** \nX
Both true if \nX > 0
.if \nX<=0, **.if** !\nX
Both true if \nX ≤ 0
.ie [expr]
If part of if...else
.el [expr]
else
.ig AX ignore till .AX
\\$1 \\$2 .../etc
parameter e.g. .fx "To" "Co" "Num" etc

Tables (use groff -t ...)

e.g. **.TS H** \"optional **H** adds a header **allbox, center, tab(;);** \"globals - term with ';' **c l r s s** \"header formatting **l l n n n.** \"5 cols. Note the '.' to terminate Title \"if .TS H was used
col1;col2;col3;col4;col5 \"';' are separators
.TE

Globals:
allbox, box, doublebox, center, expand, tab(x)
Format modifiers:
l(eft), **r**(ight), **c**(entre), **n**(umeric), **a**(lpha), **s**(pan), ^(vertical span)
Lines: **_, =**
Formats:
fCW fI i fB b
font
p+1, p8
point size
1, 2, 10 etc
column separation
w(1i) minimum column width
e equal width columns
u staggered column
z zero width entry
v3, v5 etc
vertical spacing for text blocks

...T{ must be at end of line
text block
that can
span
several lines
T}... must be at start of line

.T& Define new data format
c s s.
data

Figures (use groff -p...)

The 'current position' (cp) after .PS is at the centre of the page!!

.PS [width [height]]
starts figure
.PE ends figure
.PF end figure with 'fly-back'
line right *d*, left *d*, up *d*, down *d*, to *p* [from *q*], by (*x,y*), <-, ->, <->
box height, width, at *p*
circle rad *r*, diam *d*, at *p*, with *c*
arc rad *r*, diam *d*, cw, right *d*, left *d*, up *d*, down *d*, to *p* [from *q*], at *p*, with *c*, <-,

<table>
<tr><th>attributes</th><th>line</th><th>box</th><th>circle</th><th>arc</th><th>el</th></tr>
<tr><td>height</td><td></td><td>•</td><td></td><td></td><td>•</td></tr>
<tr><td>width</td><td></td><td>•</td><td></td><td></td><td>•</td></tr>
<tr><td>rad</td><td></td><td></td><td>•</td><td>•</td><td></td></tr>
<tr><td>diam</td><td></td><td></td><td>•</td><td>•</td><td></td></tr>
<tr><td>cw</td><td></td><td></td><td></td><td>•</td><td></td></tr>
<tr><td>right a</td><td>•</td><td></td><td></td><td>•</td><td></td></tr>
<tr><td>left a</td><td>•</td><td></td><td></td><td>•</td><td></td></tr>
<tr><td>up a</td><td>•</td><td></td><td></td><td>•</td><td></td></tr>
<tr><td>down a</td><td>•</td><td></td><td></td><td>•</td><td></td></tr>
<tr><td>to p [from q]</td><td>•</td><td></td><td></td><td>•</td><td></td></tr>
<tr><td>at p</td><td></td><td>•</td><td>•</td><td>•</td><td>•</td></tr>
<tr><td>with c</td><td></td><td></td><td>•</td><td>•</td><td>•</td></tr>
<tr><td>by (x, y)</td><td>•</td><td></td><td></td><td></td><td></td></tr>
<tr><td>dotted e</td><td>•</td><td>•</td><td>•</td><td>•</td><td>•</td></tr>
<tr><td>dashed e</td><td>•</td><td>•</td><td>•</td><td>•</td><td>•</td></tr>
<tr><td><-</td><td>•</td><td></td><td></td><td>•</td><td></td></tr>
<tr><td>-></td><td>•</td><td></td><td></td><td>•</td><td></td></tr>
<tr><td><-></td><td>•</td><td></td><td></td><td>•</td><td></td></tr>
<tr><td>"text"</td><td>•</td><td>•</td><td>•</td><td>•</td><td>•</td></tr>
<tr><td>[above|below]</td><td>•</td><td>•</td><td>•</td><td>•</td><td>•</td></tr>
<tr><td>[ljust|rjust]</td><td>•</td><td>•</td><td>•</td><td>•</td><td>•</td></tr>
</table>

->, <->

ellipse height, width, at *p*, with *c*

text All primitives can have "*text*" [above|below] [ljust|rjust]

dotted *e*, dashed *e*
All primitives may be dotted *e* or dashed *e*.

Primitives
box, circle, ellipse, arc, line, arrow, spline, *text-list*

Attributes
h(eight) *expr*, wid(th) *expr*, rad(ius) *expr*, diam(eter) *expr*, up *expr*, down *expr*, right *expr*, left *expr*, from *positon*, to *positon*, at *positon*, with *corner*, by *expr, expr*, then, dotted *expr*, dashed *expr*, chop *expr*, <-, ->, <->, invis, solid, fill *expr*, same

Text
Position:
[center ljust rjust above below]
"*text*" *position*
sprintf(*"format", expr*) *position*

Places .n .e .w .s .ne .se .nw .sw .c .start .end top bot left right start end 1st 2nd 3rd *n*th last (*x, y*)

Built-in Variables
boxwid (.75) boxht (.5) circlerad (.25) arcrad (.25) ellipsewid (.75) ellipseht (.5) linewid (.5) lineht (.5) movewid (.5) moveht (.5) textwid (0) textht (0) arrowwid (.05) arrowht (.1) dashwd (.1) arrowhead (2) maxpsht (8.5) maxpswid (11) scale (1) fillval (.3)

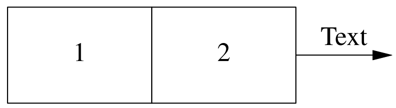
Verbs for *var* = *expr* to *expr* by *expr* do { ... }
if *expr* then { ... } else { ... }
define *name* { ... }
undef *name*

Other [...] block - names are local
{ ... } position & direction restored on exit
object-name : *elements*

example:

```
define junk {
    for i = 1 to n do {
        box sprintf("%g", i)
    }
}
n=2; Firstjunk: [junk]
arrow "Text" above  with .w \
    at Firstjunk.e
```

produces:



Equations (use groff -e)

.EQ *label*
Start equation (put into a .DS ... .DE display). *label* prints at right

delim *xx*|off
Delimiters for in-line equations (suggest $$)

.EN End of equation

sub, sup
Sub- & super-script

~, ^ Full & half space

sqrt, **over**
Simple operators

sum, **int**, **prod**, **union**, **inter**, **oppE**, **oppA**, **lim**, **max**, **min**
Operators with limits - add [**from** *x*] [**to** *x*]

left *x* ... **right** *y*
large brackets: *x*, & *y* are from (){}|
floor ceiling

define *name* **%**size +$1 bold ->**%**
Macro invoked by *name*(1). tdefine, ndefine work for troff/nroff

mark Alignment

lineup lines up to **mark**

pile Vertical alignment

matrix { **ccol** { *x* **above** *y* ... } }

example:

```
.EQ
left ( pile { n above k} right )
~=~ n! over {k! ^ ( n - k ) !}
~=~ left ( pile { n above {n - k}} \
    right )
.EN
```

produces:

$$\left(\begin{matrix} n \\ k \end{matrix}\right) = \frac{n!}{k!\,(n-k)!} = \left(\begin{matrix} n \\ n-k \end{matrix}\right)$$

Special Characters (all)

Use with \N'xxx':

128: ‚	129: «	130: »	131: •
132: ƒ	133: ⁄	134: ‰	135: †
136: ‡	137: –	138: —	139:
140: ﬁ	141: ﬂ	142:	143:
144: ı	145:	146: `	147: ˝
148: ˙	149: ˘	150: ˇ	151: ˚
152: ˛	153: “	154: ”	155: œ
156: ł	157: „	158: Œ	159: Ł
160:	161: ¡	162: ¢	163: £
164: ¤	165: ¥	166: ¦	167: §
168: ¨	169: ©	170: ª	171: ‹
172: ¬	173: –	174: ®	175: ¯
176: °	177: ±	178: ²	179: ³
180: ´	181: µ	182: ¶	183: ·
184: ¸	185: ¹	186: º	187: ›
188: ¼	189: ½	190: ¾	191: ¿
192: À	193: Á	194: Â	195: Ã
196: Ä	197: Å	198: Æ	199: Ç
200: È	201: É	202: Ê	203: Ë
204: Ì	205: Í	206: Î	207: Ï
208: Ð	209: Ñ	210: Ò	211: Ó
212: Ô	213: Õ	214: Ö	215: ×
216: Ø	217: Ù	218: Ú	219: Û
220: Ü	221: Ý	222: Þ	223: ß
224: à	225: á	226: â	227: ã
228: ä	229: å	230: æ	231: ç
232: è	233: é	234: ê	235: ë
236: ì	237: í	238: î	239: ï
240: ð	241: ñ	242: ò	243: ó
244: ô	245: õ	246: ö	247: ÷
248: ø	249: ù	250: ú	251: û
252: ü	253: ý	254: þ	255: ÿ