

## CHAPTER 8

# Higher Mathematics

Basic L<sup>A</sup>T<sub>E</sub>X offers a high level of mathematical typesetting capabilities. However, when complex equations or other mathematical constructs have to be input repeatedly, it is up to you to define new commands or environments to ease the burden of typing. The American Mathematical Society (AMS), recognizing that fact, has sponsored the development of extensions to T<sub>E</sub>X, known as  $\mathcal{A}\mathcal{M}\mathcal{S}$ -T<sub>E</sub>X. They make the preparation of mathematical compuscripts less time-consuming and the copy more consistent.

Recently these extensions were ported to L<sup>A</sup>T<sub>E</sub>X in the form of a set of packages known as “ $\mathcal{A}\mathcal{M}\mathcal{S}$ -L<sup>A</sup>T<sub>E</sub>X” [?]. As some parts of  $\mathcal{A}\mathcal{M}\mathcal{S}$ -T<sub>E</sub>X had to do with mathematics fonts the corresponding L<sup>A</sup>T<sub>E</sub>X packages went into a separate distribution called “AMSF<sub>onts</sub>”, rather than into  $\mathcal{A}\mathcal{M}\mathcal{S}$ -L<sup>A</sup>T<sub>E</sub>X.

### 8.1 The $\mathcal{A}\mathcal{M}\mathcal{S}$ -L<sup>A</sup>T<sub>E</sub>X Project

$\mathcal{A}\mathcal{M}\mathcal{S}$ -T<sub>E</sub>X was originally released for general use in 1982. Its main strength is that it facilitates mathematical typesetting, while producing output that satisfies the high standards of mathematical publishing. It provides a predefined set of natural commands such as `\matrix` and `\text` that make complicated mathematics reasonably convenient to type. These commands incorporate the typesetting experience and standards of the American Mathematical Society, to handle problematic possibilities, such as matrices within matrices or a word of text within a subscript, without burdening the user.

$\mathcal{A}\mathcal{M}\mathcal{S}$ -T<sub>E</sub>X lacks certain useful L<sup>A</sup>T<sub>E</sub>X features such as automatic numbering that adjusts to addition or deletion of material being the primary one. Nor does

it have the laborsaving abilities of L<sup>A</sup>T<sub>E</sub>X for preparing indexes, bibliographies, tables, or simple diagrams. These features are such a convenience for authors that the use of L<sup>A</sup>T<sub>E</sub>X spread rapidly in the mid-1980s (a reasonably mature version of L<sup>A</sup>T<sub>E</sub>X was available by the end of 1983), and the American Mathematical Society began to be asked by its authors to accept electronic submissions in L<sup>A</sup>T<sub>E</sub>X.

Thus, the  $\mathcal{A}\mathcal{M}\mathcal{S}$ -L<sup>A</sup>T<sub>E</sub>X project came into being in 1987 and three years later  $\mathcal{A}\mathcal{M}\mathcal{S}$ -L<sup>A</sup>T<sub>E</sub>X version 1.0 was released. The conversion of  $\mathcal{A}\mathcal{M}\mathcal{S}$ -T<sub>E</sub>X's mathematical capabilities to L<sup>A</sup>T<sub>E</sub>X, and the integration with the NFSS, were done by Frank Mittelbach and Rainer Schöpf, working as consultants to the AMS, with assistance from Michael Downes of the AMS technical support staff.

The most often used packages are `amsmath` (from  $\mathcal{A}\mathcal{M}\mathcal{S}$ -L<sup>A</sup>T<sub>E</sub>X) and `amssymb` (from the AMSFonts distribution). To invoke them in a document you write, e.g., `\usepackage{amsmath}` in the usual way. Installation and usage documentation is included with the packages. For `amssymb` the principal piece of documentation is the *AMSFonts User's Guide* (`amsfndoc.tex`); for `amsmath` it is the  *$\mathcal{A}\mathcal{M}\mathcal{S}$ -L<sup>A</sup>T<sub>E</sub>X User's Guide* (`amsldoc.tex`).<sup>1</sup>

## 8.2 Fonts and Symbols in Formulae

### 8.2.1 Mathematical Symbols

( $\mathcal{L}$  42–47)

Tables 8.2 on the next page to 8.11 on page 227 review the mathematical symbols available in standard L<sup>A</sup>T<sub>E</sub>X. You can put a slash through a L<sup>A</sup>T<sub>E</sub>X symbol by preceding it with the `\not` command, for instance.

( $\mathcal{L}$  44)

$u \not\prec v$  or  $a \notin \mathbf{A}$

`$u \not\prec v$` or `$a \not\in \mathbf{A}$`

Tables 8.12 on page 227 to 8.19 on page 229 show the extra math symbols of the  $\mathcal{A}\mathcal{M}\mathcal{S}$ -Fonts, which are automatically available when you specify the `amssymb` package.<sup>2</sup> However, if you want to define only some of them (perhaps because your T<sub>E</sub>X installation has insufficient memory to define all the symbol names), you can use the `amsfonts` package and the `\DeclareMathSymbol` command, which is explained in section 7.7.6.

<sup>1</sup> The AMS distribution also contains a file `diff12.tex` which describes differences between version 1.1 and 1.2 of  $\mathcal{A}\mathcal{M}\mathcal{S}$ -L<sup>A</sup>T<sub>E</sub>X. Note in particular that in versions 1.0 and 1.1 of  $\mathcal{A}\mathcal{M}\mathcal{S}$ -L<sup>A</sup>T<sub>E</sub>X, which predated L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>, the `amsmath` package was named “`amstex`” and included some of the font-related features that are now separated in the `amssymb` and `amsfonts` packages.

<sup>2</sup> Note that the Companion uses Lucida math fonts which contain the standard L<sup>A</sup>T<sub>E</sub>X and  $\mathcal{A}\mathcal{M}\mathcal{S}$  symbols but with different shapes compared to the Computer Modern math fonts.

$\hat{a}$	<code>\hat{a}</code>	$\acute{a}$	<code>\acute{a}</code>	$\bar{a}$	<code>\bar{a}</code>	$\dot{a}$	<code>\dot{a}</code>	$\breve{a}$	<code>\breve{a}</code>
$\check{a}$	<code>\check{a}</code>	$\grave{a}$	<code>\grave{a}</code>	$\vec{a}$	<code>\vec{a}</code>	$\ddot{a}$	<code>\ddot{a}</code>	$\tilde{a}$	<code>\tilde{a}</code>

Table 8.1: Math mode accents (available in L<sup>A</sup>T<sub>E</sub>X)

$\alpha$	<code>\alpha</code>	$\beta$	<code>\beta</code>	$\gamma$	<code>\gamma</code>	$\delta$	<code>\delta</code>	$\epsilon$	<code>\epsilon</code>
$\varepsilon$	<code>\varepsilon</code>	$\zeta$	<code>\zeta</code>	$\eta$	<code>\eta</code>	$\theta$	<code>\theta</code>	$\vartheta$	<code>\vartheta</code>
$\iota$	<code>\iota</code>	$\kappa$	<code>\kappa</code>	$\lambda$	<code>\lambda</code>	$\mu$	<code>\mu</code>	$\nu$	<code>\nu</code>
$\xi$	<code>\xi</code>	$\omicron$	<code>\omicron</code>	$\pi$	<code>\pi</code>	$\varpi$	<code>\varpi</code>	$\rho$	<code>\rho</code>
$\varrho$	<code>\varrho</code>	$\sigma$	<code>\sigma</code>	$\varsigma$	<code>\varsigma</code>	$\tau$	<code>\tau</code>	$\upsilon$	<code>\upsilon</code>
$\phi$	<code>\phi</code>	$\varphi$	<code>\varphi</code>	$\chi$	<code>\chi</code>	$\psi$	<code>\psi</code>	$\omega$	<code>\omega</code>
$\Gamma$	<code>\Gamma</code>	$\Delta$	<code>\Delta</code>	$\Theta$	<code>\Theta</code>	$\Lambda$	<code>\Lambda</code>	$\Xi$	<code>\Xi</code>
$\Pi$	<code>\Pi</code>	$\Sigma$	<code>\Sigma</code>	$\Upsilon$	<code>\Upsilon</code>	$\Phi$	<code>\Phi</code>	$\Psi$	<code>\Psi</code>
$\Omega$	<code>\Omega</code>								

Table 8.2: Greek letters (available in L<sup>A</sup>T<sub>E</sub>X)

$\pm$	<code>\pm</code>	$\cap$	<code>\cap</code>	$\diamond$	<code>\diamond</code>	$\oplus$	<code>\oplus</code>
$\mp$	<code>\mp</code>	$\cup$	<code>\cup</code>	$\triangleup$	<code>\triangleup</code>	$\ominus$	<code>\ominus</code>
$\times$	<code>\times</code>	$\uplus$	<code>\uplus</code>	$\triangledown$	<code>\triangledown</code>	$\otimes$	<code>\otimes</code>
$\div$	<code>\div</code>	$\sqcap$	<code>\sqcap</code>	$\triangleleft$	<code>\triangleleft</code>	$\oslash$	<code>\oslash</code>
$*$	<code>\ast</code>	$\sqcup$	<code>\sqcup</code>	$\triangleright$	<code>\triangleright</code>	$\odot$	<code>\odot</code>
$\star$	<code>\star</code>	$\vee$	<code>\vee</code>	$\triangleleft^a$	<code>\lhd^a</code>	$\bigcirc$	<code>\bigcirc</code>
$\circ$	<code>\circ</code>	$\wedge$	<code>\wedge</code>	$\triangleright^a$	<code>\rhd^a</code>	$\dagger$	<code>\dagger</code>
$\bullet$	<code>\bullet</code>	$\setminus$	<code>\setminus</code>	$\triangleleft^a$	<code>\unlhd^a</code>	$\ddagger$	<code>\ddagger</code>
$\cdot$	<code>\cdot</code>	$\wr$	<code>\wr</code>	$\triangleright^a$	<code>\unrhd^a</code>	$\amalg$	<code>\amalg</code>

<sup>a</sup> Not predefined in NFSS. Use the `latexsym` or `amssymb` package.

Table 8.3: Binary operation symbols (available in L<sup>A</sup>T<sub>E</sub>X)

$\leq$	<code>\leq</code> , <code>\le</code>	$\geq$	<code>\geq</code> , <code>\ge</code>	$\equiv$	<code>\equiv</code>	$\models$	<code>\models</code>	$\prec$	<code>\prec</code>
$\succ$	<code>\succ</code>	$\sim$	<code>\sim</code>	$\perp$	<code>\perp</code>	$\preceq$	<code>\preceq</code>	$\succeq$	<code>\succeq</code>
$\simeq$	<code>\simeq</code>	$\mid$	<code>\mid</code>	$\ll$	<code>\ll</code>	$\gg$	<code>\gg</code>	$\asymp$	<code>\asymp</code>
$\parallel$	<code>\parallel</code>	$\subset$	<code>\subset</code>	$\supset$	<code>\supset</code>	$\approx$	<code>\approx</code>	$\bowtie$	<code>\bowtie</code>
$\subseteq$	<code>\subseteq</code>	$\supseteq$	<code>\supseteq</code>	$\cong$	<code>\cong</code>	$\Join$	<code>\Join</code>	$\sqsubset$	<code>\sqsubset</code>
$\sqsupseteq$	<code>\sqsupseteq</code>	$\neq$	<code>\neq</code>	$\smile$	<code>\smile</code>	$\sqsubseteq$	<code>\sqsubseteq</code>	$\sqsupseteq$	<code>\sqsupseteq</code>
$\doteq$	<code>\doteq</code>	$\frown$	<code>\frown</code>	$\in$	<code>\in</code>	$\ni$	<code>\ni</code>	$\propto$	<code>\propto</code>
$=$	<code>=</code>	$\vdash$	<code>\vdash</code>	$\dashv$	<code>\dashv</code>	$<$	<code>&lt;</code>	$>$	<code>&gt;</code>

Table 8.4: Relation symbols (available in L<sup>A</sup>T<sub>E</sub>X)

Chapter 8 of "The L<sup>A</sup>T<sub>E</sub>X Companion", updated for AMS-L<sup>A</sup>T<sub>E</sub>X version 1.2 (Sep. 1st 1997).

Copyright © 1997 by Addison Wesley Longman, Inc. All rights reserved.

$\leftarrow$	<code>\leftarrow</code>	$\longleftarrow$	<code>\longleftarrow</code>	$\uparrow$	<code>\uparrow</code>
$\Leftarrow$	<code>\Leftarrow</code>	$\Longleftarrow$	<code>\Longleftarrow</code>	$\Uparrow$	<code>\Uparrow</code>
$\rightarrow$	<code>\rightarrow</code>	$\longrightarrow$	<code>\longrightarrow</code>	$\downarrow$	<code>\downarrow</code>
$\Rightarrow$	<code>\Rightarrow</code>	$\Longrightarrow$	<code>\Longrightarrow</code>	$\Downarrow$	<code>\Downarrow</code>
$\leftrightarrow$	<code>\leftrightarrow</code>	$\longleftrightarrow$	<code>\longleftrightarrow</code>	$\updownarrow$	<code>\updownarrow</code>
$\Leftrightarrow$	<code>\Leftrightarrow</code>	$\Longleftrightarrow$	<code>\Longleftrightarrow</code>	$\Updownarrow$	<code>\Updownarrow</code>
$\mapsto$	<code>\mapsto</code>	$\longmapsto$	<code>\longmapsto</code>	$\nearrow$	<code>\nearrow</code>
$\hookrightarrow$	<code>\hookrightarrow</code>	$\hookleftarrow$	<code>\hookleftarrow</code>	$\searrow$	<code>\searrow</code>
$\lhookrightarrow$	<code>\lhookrightarrow</code>	$\rhookrightarrow$	<code>\rhookrightarrow</code>	$\swarrow$	<code>\swarrow</code>
$\leftharpoonup$	<code>\leftharpoonup</code>	$\rightharpoonup$	<code>\rightharpoonup</code>	$\nwarrow$	<code>\nwarrow</code>
$\leftharpoondown$	<code>\leftharpoondown</code>	$\rightharpoondown$	<code>\rightharpoondown</code>		

Table 8.5: Arrow symbols (available in L<sup>A</sup>T<sub>E</sub>X)

$\dots$	<code>\ldots</code>	$\cdots$	<code>\cdots</code>	$\vdots$	<code>\vdots</code>	$\ddots$	<code>\ddots</code>	$\aleph$	<code>\aleph</code>
$\prime$	<code>\prime</code>	$\forall$	<code>\forall</code>	$\infty$	<code>\infty</code>	$\hbar$	<code>\hbar</code>	$\emptyset$	<code>\emptyset</code>
$\exists$	<code>\exists</code>	$\nabla$	<code>\nabla</code>	$\sqrt{\quad}$	<code>\sqrt{\quad}</code>	$\square$	<code>\Box<sup>a</sup></code>	$\triangle$	<code>\triangle</code>
$\diamond$	<code>\Diamond<sup>a</sup></code>	$\imath$	<code>\imath</code>	$\jmath$	<code>\jmath</code>	$\ell$	<code>\ell</code>	$\neg$	<code>\neg</code>
$\top$	<code>\top</code>	$\flat$	<code>\flat</code>	$\natural$	<code>\natural</code>	$\sharp$	<code>\sharp</code>	$\wp$	<code>\wp</code>
$\perp$	<code>\perp</code>	$\clubsuit$	<code>\clubsuit</code>	$\diamondsuit$	<code>\diamondsuit</code>	$\heartsuit$	<code>\heartsuit</code>	$\spadesuit$	<code>\spadesuit</code>
$\Uparrow$	<code>\mho<sup>a</sup></code>	$\Re$	<code>\Re</code>	$\Im$	<code>\Im</code>	$\angle$	<code>\angle</code>	$\partial$	<code>\partial</code>

<sup>a</sup> Not predefined in NFSS. Use the latexsym or amssymb package.

Table 8.6: Miscellaneous symbols (available in L<sup>A</sup>T<sub>E</sub>X)

$\sum$	<code>\sum</code>	$\prod$	<code>\prod</code>	$\coprod$	<code>\coprod</code>	$\int$	<code>\int</code>	$\oint$	<code>\oint</code>
$\bigcap$	<code>\bigcap</code>	$\bigcup$	<code>\bigcup</code>	$\bigsqcup$	<code>\bigsqcup</code>	$\bigvee$	<code>\bigvee</code>	$\bigwedge$	<code>\bigwedge</code>
$\bigodot$	<code>\bigodot</code>	$\bigotimes$	<code>\bigotimes</code>	$\bigoplus$	<code>\bigoplus</code>	$\biguplus$	<code>\biguplus</code>		

Table 8.7: Variable-sized symbols (available in L<sup>A</sup>T<sub>E</sub>X)

<code>\arccos</code>	<code>\cos</code>	<code>\csc</code>	<code>\exp</code>	<code>\ker</code>	<code>\limsup</code>	<code>\min</code>	<code>\sinh</code>
<code>\arcsin</code>	<code>\cosh</code>	<code>\deg</code>	<code>\gcd</code>	<code>\lg</code>	<code>\ln</code>	<code>\Pr</code>	<code>\sup</code>
<code>\arctan</code>	<code>\cot</code>	<code>\det</code>	<code>\hom</code>	<code>\lim</code>	<code>\log</code>	<code>\sec</code>	<code>\tan</code>
<code>\arg</code>	<code>\coth</code>	<code>\dim</code>	<code>\inf</code>	<code>\liminf</code>	<code>\max</code>	<code>\sin</code>	<code>\tanh</code>

Table 8.8: Log-like symbols (available in L<sup>A</sup>T<sub>E</sub>X)

$\uparrow$	<code>\uparrow</code>	$\Uparrow$	<code>\Uparrow</code>	$\downarrow$	<code>\downarrow</code>	$\Downarrow$	<code>\Downarrow</code>
$\{$	<code>\{</code>	$\}$	<code>\}</code>	$\updownarrow$	<code>\updownarrow</code>	$\Updownarrow$	<code>\Updownarrow</code>
$\lfloor$	<code>\lfloor</code>	$\rfloor$	<code>\rfloor</code>	$\lceil$	<code>\lceil</code>	$\rceil$	<code>\rceil</code>
$\langle$	<code>\langle</code>	$\rangle$	<code>\rangle</code>	$/$	<code>/</code>	$\backslash$	<code>\backslash</code>
$ $	<code> </code>	$\ $	<code>\ </code>				

Table 8.9: Delimiters (available in L<sup>A</sup>T<sub>E</sub>X)

$\left\{$	<code>\rmoustache</code>	$\int$	<code>\lmoustache</code>	$\right\}$	<code>\rgroup</code>	$\left($	<code>\lgroup</code>
$ $	<code>\arrowvert</code>	$\ $	<code>\Arrowvert</code>	$ $	<code>\bracevert</code>		

Table 8.10: Large delimiters (available in L<sup>A</sup>T<sub>E</sub>X)

$\widetilde{abc}$	<code>\widetilde{abc}</code>	$\widehat{abc}$	<code>\widehat{abc}</code>
$\overleftarrow{abc}$	<code>\overleftarrow{abc}</code>	$\overrightarrow{abc}$	<code>\overrightarrow{abc}</code>
$\overline{abc}$	<code>\overline{abc}</code>	$\underline{abc}$	<code>\underline{abc}</code>
$\overbrace{abc}$	<code>\overbrace{abc}</code>	$\underbrace{abc}$	<code>\underbrace{abc}</code>
$\sqrt{abc}$	<code>\sqrt{abc}</code>	$\sqrt[n]{abc}$	<code>\sqrt[n]{abc}</code>
$f'$	<code>f'</code>	$\frac{abc}{xyz}$	<code>\frac{abc}{xyz}</code>

Table 8.11: L<sup>A</sup>T<sub>E</sub>X math constructs

$\digamma$	<code>\digamma</code>	$\varkappa$	<code>\varkappa</code>	$\beth$	<code>\beth</code>	$\daleth$	<code>\daleth</code>	$\gimel$	<code>\gimel</code>
------------	-----------------------	-------------	------------------------	---------	--------------------	-----------	----------------------	----------	---------------------

Table 8.12: AMS Greek and Hebrew (available with amssymb package)

$\ulcorner$	<code>\ulcorner</code>	$\urcorner$	<code>\urcorner</code>	$\llcorner$	<code>\llcorner</code>	$\lrcorner$	<code>\lrcorner</code>
-------------	------------------------	-------------	------------------------	-------------	------------------------	-------------	------------------------

Table 8.13: AMS delimiters (available with amssymb package)

$\Rightarrow$	<code>\Rrightarrow</code>	$\rightsquigarrow$	<code>\rightsquigarrow</code>	$\Leftleftarrows$	<code>\leftleftarrows</code>
$\Leftrightarrow$	<code>\Leftrightarrow</code>	$\Lleftarrow$	<code>\Lleftarrow</code>	$\twoheadleftarrow$	<code>\twoheadleftarrow</code>
$\leftarrowtail$	<code>\leftarrowtail</code>	$\looparrowleft$	<code>\looparrowleft</code>	$\leftrightharpoons$	<code>\leftrightharpoons</code>
$\curvearrowleft$	<code>\curvearrowleft</code>	$\circlearrowleft$	<code>\circlearrowleft</code>	$\Lsh$	<code>\Lsh</code>
$\Uparrow$	<code>\Uparrow</code>	$\upharpoonleft$	<code>\upharpoonleft</code>	$\Downharpoonleft$	<code>\Downharpoonleft</code>
$\multimap$	<code>\multimap</code>	$\leftrightsquigarrow$	<code>\leftrightsquigarrow</code>	$\rightleftarrows$	<code>\rightleftarrows</code>
$\Rrightarrowtail$	<code>\Rrightarrowtail</code>	$\twoheadrightarrow$	<code>\twoheadrightarrow</code>	$\rightarrowtail$	<code>\rightarrowtail</code>
$\looparrowright$	<code>\looparrowright</code>	$\rightleftharpoons$	<code>\rightleftharpoons</code>	$\curvearrowright$	<code>\curvearrowright</code>
$\circlearrowright$	<code>\circlearrowright</code>	$\Rsh$	<code>\Rsh</code>	$\Downarrow$	<code>\Downarrow</code>
$\Downharpoonright$	<code>\Downharpoonright</code>	$\upharpoonright$	<code>\upharpoonright</code>	$\restriction$	<code>\restriction</code>

Table 8.14: AMS arrows (available with amssymb package)

$\nleftarrow$	<code>\nleftarrow</code>	$\nrightarrow$	<code>\nrightarrow</code>	$\nLeftarrow$	<code>\nLeftarrow</code>
$\nrightarrowtail$	<code>\nrightarrowtail</code>	$\nlefttrightarrow$	<code>\nlefttrightarrow</code>	$\nLefttrightarrow$	<code>\nLefttrightarrow</code>

Table 8.15: AMS negated arrows (available with amssymb package)

$\leqq$	<code>\leqq</code>	$\leqslant$	<code>\leqslant</code>	$\leqslantless$	<code>\leqslantless</code>
$\lesssim$	<code>\lesssim</code>	$\lessapprox$	<code>\lessapprox</code>	$\approxex$	<code>\approxex</code>
$\lessdot$	<code>\lessdot</code>	$\lll, \lllless$	<code>\lll, \lllless</code>	$\lessgtr$	<code>\lessgtr</code>
$\lesseqgtr$	<code>\lesseqgtr</code>	$\lesseqqgtr$	<code>\lesseqqgtr</code>	$\doteqdot, \Doteq$	<code>\doteqdot, \Doteq</code>
$\risingdotseq$	<code>\risingdotseq</code>	$\fallingdotseq$	<code>\fallingdotseq</code>	$\backsim$	<code>\backsim</code>
$\backsimeq$	<code>\backsimeq</code>	$\subseteq$	<code>\subseteq</code>	$\Subset$	<code>\Subset</code>
$\sqsubset$	<code>\sqsubset</code>	$\prec$	<code>\prec</code>	$\curlyeqprec$	<code>\curlyeqprec</code>
$\precsim$	<code>\precsim</code>	$\precapprox$	<code>\precapprox</code>	$\triangleleft$	<code>\triangleleft</code>
$\trianglelefteq$	<code>\trianglelefteq</code>	$\vDash$	<code>\vDash</code>	$\Vdash$	<code>\Vdash</code>
$\smallsmile$	<code>\smallsmile</code>	$\smallfrown$	<code>\smallfrown</code>	$\bumpeq$	<code>\bumpeq</code>
$\Bumpeq$	<code>\Bumpeq</code>	$\geqq$	<code>\geqq</code>	$\geqslant$	<code>\geqslant</code>
$\eqslantgtr$	<code>\eqslantgtr</code>	$\gtrsim$	<code>\gtrsim</code>	$\gtrapprox$	<code>\gtrapprox</code>
$\gtrdot$	<code>\gtrdot</code>	$\ggg, \gggtr$	<code>\ggg, \gggtr</code>	$\gtrless$	<code>\gtrless</code>
$\gtreqless$	<code>\gtreqless</code>	$\gtreqqless$	<code>\gtreqqless</code>	$\eqcirc$	<code>\eqcirc</code>
$\circeq$	<code>\circeq</code>	$\triangleq$	<code>\triangleq</code>	$\thicksim$	<code>\thicksim</code>
$\thickapprox$	<code>\thickapprox</code>	$\supseteq$	<code>\supseteq</code>	$\Supset$	<code>\Supset</code>
$\sqsupset$	<code>\sqsupset</code>	$\succ$	<code>\succ</code>	$\curlyeqsucc$	<code>\curlyeqsucc</code>
$\succsim$	<code>\succsim</code>	$\succapprox$	<code>\succapprox</code>	$\vartriangleright$	<code>\vartriangleright</code>
$\trianglerighteq$	<code>\trianglerighteq</code>	$\VDash$	<code>\VDash</code>	$\shortmid$	<code>\shortmid</code>
$\shortparallel$	<code>\shortparallel</code>	$\between$	<code>\between</code>	$\pitchfork$	<code>\pitchfork</code>
$\varpropto$	<code>\varpropto</code>	$\blacktriangleleft$	<code>\blacktriangleleft</code>	$\therefore$	<code>\therefore</code>
$\backepsilon$	<code>\backepsilon</code>	$\blacktriangleright$	<code>\blacktriangleright</code>	$\because$	<code>\because</code>

Table 8.16: AMS binary relations (available with amssymb package)

$\nless$	<code>\nless</code>	$\nleq$	<code>\nleq</code>	$\nleqslant$	<code>\nleqslant</code>
$\nleqq$	<code>\nleqq</code>	$\nleq$	<code>\nleq</code>	$\nleqq$	<code>\nleqq</code>
$\nvertneqq$	<code>\nvertneqq</code>	$\nlsim$	<code>\nlsim</code>	$\nlnapprox$	<code>\nlnapprox</code>
$\nprec$	<code>\nprec</code>	$\npreceq$	<code>\npreceq</code>	$\nprecnsim$	<code>\nprecnsim</code>
$\nprecapprox$	<code>\nprecapprox</code>	$\nsim$	<code>\nsim</code>	$\nshortmid$	<code>\nshortmid</code>
$\nmid$	<code>\nmid</code>	$\nvdash$	<code>\nvdash</code>	$\nvDash$	<code>\nvDash</code>
$\ntriangleleft$	<code>\ntriangleleft</code>	$\ntrianglelefteq$	<code>\ntrianglelefteq</code>	$\nsubseteq$	<code>\nsubseteq</code>
$\nsubseteq$	<code>\nsubseteq</code>	$\nvarsubsetneq$	<code>\nvarsubsetneq</code>	$\nsubsetneqq$	<code>\nsubsetneqq</code>
$\nvarsubsetneqq$	<code>\nvarsubsetneqq</code>	$\ngtr$	<code>\ngtr</code>	$\ngeq$	<code>\ngeq</code>
$\ngeqslant$	<code>\ngeqslant</code>	$\ngeqq$	<code>\ngeqq</code>	$\gneq$	<code>\gneq</code>
$\gneqq$	<code>\gneqq</code>	$\gvertneqq$	<code>\gvertneqq</code>	$\gnsim$	<code>\gnsim</code>
$\gnapprox$	<code>\gnapprox</code>	$\nsucc$	<code>\nsucc</code>	$\nsucceq$	<code>\nsucceq</code>
$\succnsim$	<code>\succnsim</code>	$\succapprox$	<code>\succapprox</code>	$\ncong$	<code>\ncong</code>
$\nshortparallel$	<code>\nshortparallel</code>	$\nparallel$	<code>\nparallel</code>	$\nvDash$	<code>\nvDash</code>
$\nVDash$	<code>\nVDash</code>	$\ntriangleright$	<code>\ntriangleright</code>	$\ntrianglerighteq$	<code>\ntrianglerighteq</code>
$\nsupseteq$	<code>\nsupseteq</code>	$\nsupseteqq$	<code>\nsupseteqq</code>	$\supsetneq$	<code>\supsetneq</code>
$\nvarsupsetneq$	<code>\nvarsupsetneq</code>	$\supsetneqq$	<code>\supsetneqq</code>	$\varsupsetneqq$	<code>\varsupsetneqq</code>

Table 8.17: AMS negated binary relations (available with amssymb package)

$\dot{+}$	<code>\dotplus</code>	$\smallsetminus$	<code>\smallsetminusminus</code>	$\Cap$	<code>\Cap,\doublecap</code>
$\Cup$	<code>\Cup,\doublecup</code>	$\bar{\wedge}$	<code>\barwedge</code>	$\veebar$	<code>\veebar</code>
$\overline{\wedge}$	<code>\doublebarwedge</code>	$\boxminus$	<code>\boxminus</code>	$\boxtimes$	<code>\boxtimes</code>
$\boxdot$	<code>\boxdot</code>	$\boxplus$	<code>\boxplus</code>	$\div$	<code>\divideontimes</code>
$\ltimes$	<code>\ltimes</code>	$\rtimes$	<code>\rtimes</code>	$\leftthreetimes$	<code>\leftthreetimes</code>
$\rightthreetimes$	<code>\rightthreetimes</code>	$\curlywedge$	<code>\curlywedge</code>	$\curlyvee$	<code>\curlyvee</code>
$\circleddash$	<code>\circleddash</code>	$\circledast$	<code>\circledast</code>	$\circledcirc$	<code>\circledcirc</code>
$\centerdot$	<code>\centerdot</code>	$\intercal$	<code>\intercal</code>		

Table 8.18: AMS binary operators (available with `amssymb` package)

$\hbar$	<code>\hbar</code>	$\hslash$	<code>\hslash</code>	$\vartriangle$	<code>\vartriangle</code>
$\nabla$	<code>\triangledown</code>	$\square$	<code>\square</code>	$\lozenge$	<code>\lozenge</code>
$\textcircled{S}$	<code>\circledS</code>	$\sphericalangle$	<code>\angle</code>	$\sphericalangle$	<code>\measuredangle</code>
$\nexists$	<code>\nexists</code>	$\mho$	<code>\mho</code>	$\Finv$	<code>\Finv</code>
$\complement$	<code>\Game</code>	$\mathbb{k}$	<code>\Bbbk</code>	$\backprime$	<code>\backprime</code>
$\varnothing$	<code>\varnothing</code>	$\blacktriangle$	<code>\blacktriangle</code>	$\blacktriangledown$	<code>\blacktriangledown</code>
$\blacksquare$	<code>\blacksquare</code>	$\blacklozenge$	<code>\blacklozenge</code>	$\bigstar$	<code>\bigstar</code>
$\sphericalangle$	<code>\sphericalangle</code>	$\complement$	<code>\complement</code>	$\eth$	<code>\eth</code>
$\diagup$	<code>\diagup</code>	$\diagdown$	<code>\diagdown</code>		

Table 8.19: AMS miscellaneous (available with `amssymb` package)

## 8.2.2 Names of Math Font Commands

The list of math font commands provided by the  $\mathcal{AMS}$  packages is shown in table 8.20 on the next page, where for each case an example is shown. In addition, the math font commands of table 7.4 on page 183 can be used.

In the `amsmath` package, `\boldsymbol` is to be used for individual bold math symbols and bold Greek letters—everything in math except for letters (where one would use `\mathbf`). For example, to obtain a bold  $\infty$ , or `\boldsymbol{\infty}`, `\boldsymbol{+}`, `\boldsymbol{\pi}`, or `\boldsymbol{0}`.

Since `\boldsymbol` takes a lot of typing, you can introduce new commands for bold symbols to be used frequently:

$B_\infty + \pi B_1 \sim \mathbf{B}_\infty + \pi \mathbf{B}_1$	<pre> \newcommand{\bpi}{\boldsymbol{\pi}} \newcommand{\binfty}{\boldsymbol{\infty}} \[ B_\infty + \pi B_1 \sim \mathbf{B}_\infty + \pi \mathbf{B}_1 \] </pre>
--	---

For those math symbols where the command `\boldsymbol` has no effect because the bold version of the symbol does not exist in the currently available fonts, there exists a command “Poor man’s bold” (`\pmb`), which simulates bold

<code>\mathbb</code>	Blackboard bold alphabet, e.g., <code>\mathbb{NQRZ}</code> gives: $\mathbb{NQRZ}$ (not available in <code>amsmath</code> , need to load <code>amssymb</code> ).
<code>\mathfrak</code>	Euler Fraktur alphabet, e.g., <code>\mathfrak{E}=\mathfrak{mc}^2</code> gives: $\mathfrak{E} = \mathfrak{mc}^2$ (not available in <code>amsmath</code> , need to load <code>amssymb</code> ).
<code>\boldsymbol</code>	Used to obtain bold numbers and other nonalphabetic symbols, as well as bold Greek letters (defined in <code>amsbsy</code> ).
<code>\pmb</code>	“Poor man’s bold,” used for math symbols when bold versions don’t exist in the available fonts, e.g., <code>\pmb{\oint}</code> gives: $\oint$ and <code>\pmb{\triangle}</code> gives: $\triangle$ (defined in <code>amsbsy</code> ).
<code>\text</code>	Produce normal text with correct text-spacing in the current font used outside math, e.g., <code>\text{E=mc}^2\text{\text{(Einstein)}}</code> gives: $E = mc^2$ (Einstein) (defined in <code>amstext</code> ).

Table 8.20: Font commands available in mathematics with the  $\mathcal{AMS}$  packages

by typesetting several copies of the symbol with slight offsets. This procedure must be used for the extension and large operator symbols from the `cmex` font, as well as the  $\mathcal{AMS}$  extra math symbols from the `msam` and `msbm` fonts.

$$\frac{\partial w}{\partial u} \bigg| \frac{\partial u}{\partial v} \quad \begin{array}{l} \backslash [ \ \frac{\partial w}{\partial u} \bigg| \frac{\partial u}{\partial v} \\ \quad \backslash \text{Bigg}\text{vert} \\ \quad \backslash \text{frac}\{\partial w\}\{\partial u\} \backslash \end{array}$$

With large operators and extension symbols (for example,  $\sum$  and  $\prod$ ) `\pmb` does not currently work very well because the proper spacing and treatment of limits is not preserved. Therefore, the  $\text{\TeX}$  operator `\mathop` needs to be used (see table 7.13 on page 213).

$$\sum_{j < P} \prod_{\lambda} \lambda R(r_i) \quad \sum_{x_j} \prod_{\lambda} \lambda R(x_j) \quad \begin{array}{l} \backslash [ \ \sum_{j < P} \\ \quad \prod_{\lambda} \lambda R(r_i) \ \text{qqquad} \\ \quad \mathop{\pmb{\sum}}_{x_j} \\ \quad \mathop{\pmb{\prod}}_{\lambda} \lambda R(x_j) \\ \backslash ] \end{array}$$

To make an entire math formula bold (or as much of it as possible, depending on the available fonts), use `\boldmath` preceding the formula.

The sequence `\mathbf{\hat{A}}` produces a bold accent character over the **A**. However, combinations like `\mathcal{\hat{A}}` will not work in ordinary  $\text{\TeX}$  because the `\mathcal` font does not have its own accents. In the `amsmath` package the font change commands are defined in such a way that accent characters will be taken from the `\mathrm` font if they are not available in the current font (in addition to the `\mathcal` font, the `\mathbb` and `\mathfrak` fonts don’t contain accents).

## 8.3 Compound Symbols, Delimiters, Operators

This section<sup>3</sup> presents the math commands that are available through the `amsmath` package, which supplements L<sup>A</sup>T<sub>E</sub>X in the area of compound symbols, large delimiters, etc. In the examples, `amsmath`'s alignment environments are used. In principle a detailed understanding of how they work is not necessary at this stage, but an interested reader can turn to section 8.5 for more information.

### 8.3.1 Multiple Integral Signs

`\iint`, `\iiint`, and `\iiint` give multiple integral signs, with the spacing between them nicely adjusted, in both text and display style. `\idotsint` gives two integral signs with dots between them.

$$\iint_V \mu(u, v) \, du \, dv \quad (8.1)$$

$$\iiint_V \mu(u, v, w) \, du \, dv \, dw \quad (8.2)$$

$$\iiint_V \mu(t, u, v, w) \, dt \, du \, dv \, dw \quad (8.3)$$

$$\int \cdots \int_V \mu(u_1, \dots, u_k) \quad (8.4)$$

```
\begin{gather}
\iint\limits_V \mu(u,v)\,du\,dv \quad \backslash\backslash
\iiint\limits_V \mu(u,v,w)\,du\,dv\,dw \quad \backslash\backslash
\iiint\limits_V \mu(t,u,v,w)\,dt\,du\,dv\,dw \quad \backslash\backslash
\idotsint\limits_V \mu(u_1,\dots,u_k)
\end{gather}
```

### 8.3.2 Over and Under Arrows

Some extra over and under arrow operations are available. (In standard L<sup>A</sup>T<sub>E</sub>X one has `\overrightarrow` and `\overleftarrow`.)

$\overrightarrow{\psi_\delta(t)E_t h}$	$\psi_\delta(t)E_t h$	<code>\begin{align*}</code>
$\overleftarrow{\psi_\delta(t)E_t h}$	$\psi_\delta(t)E_t h$	<code>\overrightarrow{\psi_\delta(t) E_t h} \quad \&amp;</code>
$\overleftrightarrow{\psi_\delta(t)E_t h}$	$\psi_\delta(t)E_t h$	<code>=\underrightarrow{\psi_\delta(t) E_t h} \quad \backslash\backslash</code>
$\overleftarrow{\psi_\delta(t)E_t h}$	$\psi_\delta(t)E_t h$	<code>\overleftarrow{\psi_\delta(t) E_t h} \quad \&amp;</code>
$\overleftrightarrow{\psi_\delta(t)E_t h}$	$\psi_\delta(t)E_t h$	<code>=\underleftarrow{\psi_\delta(t) E_t h} \quad \backslash\backslash</code>
		<code>\overleftrightharrow{\psi_\delta(t) E_t h} \quad \&amp;</code>
		<code>=\underleftrightharrow{\psi_\delta(t) E_t h}</code>
		<code>\end{align*}</code>

These arrows all scale properly in subscript sizes, as seen in the following integral  $\int_{\overrightarrow{uv}} vt \, dt$ , which was coded as `\int_{\overrightarrow{uv}} vt\,dt`.

<sup>3</sup> Some material in this and the following sections is reprinted from the electronic document `testmath.tex` (distributed with `AMS-LATEX`) with permission of the American Mathematical Society.

### 8.3.3 Dots

Ellipsis dots should almost always be typed as `\dots`. Positioning (on the baseline or centered) is automatically selected according to whatever follows the `\dots`. If the next character is a plus sign, the dots will be centered; if it's a comma, they will be on the baseline. These default dot placements provided by the `amsmath` package can be changed if different conventions are wanted.

If the dots fall at the end of a math formula, the next character will be something like `\end` or `\)` or `$`, which does not give any information about how to place the dots. If that is the case, you must help by using `\dotsc` for “dots with commas,” or `\dotsb` for “dots with binary operators/relations,” or `\dotsm` for “multiplication dots,” or `\dotsi` for “dots with integrals.” In the example below, low dots are produced in the first instance and centered dots in the others, with the spacing on either side of the dots nicely adjusted.

A series  $H_1, H_2, \dots$ , a regional sum  $H_1 + H_2 + \dots$ , an orthogonal product  $H_1 H_2 \dots$ , and an infinite integral

$$\int_{H_1} \int_{H_2} \dots$$

A series `$H_1, H_2, \dotsc$`,  
a regional sum `$H_1 + H_2 + \dotsb$`, an  
orthogonal product `$H_1 H_2 \dotsm$`, and  
an infinite integral  
`\[\int_{H_1} \int_{H_2} \dotsi\]`.

### 8.3.4 Accents in Math

The following accent commands automatically position double accents correctly:

$\acute{A}$	$\bar{\bar{B}}$	$\check{C}$	$\ddot{D}$	<pre>\begin{gather*} \Acute{\Acute{A}} \quad \Bar{\Bar{B}} \quad \quad \Breve{\Breve{C}} \quad \quad \Check{\Check{D}} \quad \backslash\backslash \Ddot{\Ddot{E}} \quad \quad \Dot{\Dot{F}} \quad \quad \quad \quad \Grave{\Grave{G}} \quad \quad \Hat{\Hat{H}} \quad \quad \quad \backslash\backslash \Tilde{\Tilde{I}} \quad \quad \Vec{\Vec{J}} \end{gather*}</pre>
$\ddot{E}$	$\dot{F}$	$\grave{G}$	$\hat{H}$	
	$\tilde{I}$	$\vec{J}$		

This double accent operation is complicated and tends to slow down the processing of a `LATEX` file. If the document contains many double accents, you can load the `amsxtra` package. It defines the `\accentedsymbol` command, which you can use in the preamble of your document to help speed things up. It stores the result of the double accent command in a box register for quick retrieval. `\accentedsymbol` is used like `\newcommand`:

This is a double hat  $\hat{\hat{A}}$  and this  $\dot{\bar{\delta}}$  a delta with a bar and a dot.

```
\accentedsymbol{\Ahathat}{\Hat{\Hat A}}
\accentedsymbol{\dbardot}{\Dot{\Bar \delta}}
This is a double hat \(\Ahathat\) and this
\(\dbardot\) a delta with a bar and a dot.
```

### 8.3.5 Superscripted Accents

Some accents have a wide form: typing `\widehat{xy}`, `\widetilde{xy}` produces  $\widehat{xy}$ ,  $\widetilde{xy}$ . Because these wide accents have a certain maximum size, the `amsxtra` package introduces a different notation to handle extremely long expressions:  $(AmBD)^{\widehat{\phantom{x}}}$  instead of  $\widehat{AmBD}$ . `amsxtra` has the following control sequences to achieve this easily:

$(AmBD)^{\widehat{\phantom{x}}}$	$(AmBD)^{\spcheck}$	(8.5)	<code>\begin{gather}</code>
			<code>(AmBD)\sphat \quad \quad (AmBD)\spcheck \quad \backslash\backslash</code>
$(AmBD)^{\widetilde{\phantom{x}}}$	$(AmBD)^{\sptilde}$	(8.6)	<code>(AmBD)\sptilde \quad \quad (AmBD)\spdot \quad \backslash\backslash</code>
$(AmBD)^{\spddot}$	$(AmBD)^{\spdddot}$	(8.7)	<code>(AmBD)\spddot \quad \quad (AmBD)\spdddot \quad \backslash\backslash</code>
	$(AmBD)^{\spbreve}$	(8.8)	<code>(AmBD)\spbreve</code>
			<code>\end{gather}</code>

### 8.3.6 Dot Accents

`\dddot` and `\ddddot` are available to produce tripled and quadrupled dot accents in addition to the `\dot` and `\ddot` accents already available in L<sup>A</sup>T<sub>E</sub>X:

$\ddot{Q}$	$\ddot{R}$	<code>\dddot{Q}</code>	<code>\quad</code>	<code>\ddddot{R}</code>	<code>\$</code>
------------	------------	------------------------	--------------------	-------------------------	-----------------

### 8.3.7 Roots

In ordinary L<sup>A</sup>T<sub>E</sub>X the placement of root indices is sometimes not good. With `amsmath` the commands `\leftroot` and `\uproot` allow the adjustment of the position of the root. Positive arguments to these commands will move the root index to the left and up respectively, while a negative argument will move them right and down. The units of increment are quite small, which is useful for such adjustments. In the example below, the root index  $\beta$  is moved 2 units to the left and 4 units up.

$\sqrt[\beta]{k}$	$\sqrt[\beta]{k}$	<code>\[ \sqrt[\beta]{k} \quad \quad \quad</code>
		<code>\sqrt[\leftroot{2}\uproot{4}\beta]{k} \quad \backslash]</code>

### 8.3.8 Boxed Formulae

The command `\boxed` puts a box around its argument, similar to `\fbox`, except that the contents are in math mode:

$\boxed{W_t - F \subseteq V(P_i) \subseteq W_t}$	<code>\[ \boxed{W_t - F \subseteq V(P_i) \subseteq W_t} \quad \backslash]</code>
--	--

### 8.3.9 Extensible Arrows

`\xleftarrow` and `\xrightarrow` produce arrows that extend automatically to accommodate unusually wide subscripts or superscripts. The text of the subscript or superscript are given as an optional and mandatory argument, respectively:

$$0 \xleftarrow[\zeta]{\alpha} F \times \Delta[n-1] \xrightarrow{\partial_0 \alpha(b)} E^{\partial_0 b}$$

```

\left[ \xleftarrow[\zeta]{\alpha}
      F \times \triangle[n-1]
      \xrightarrow{\partial_0 \alpha(b)}
      E^{\partial_0 b}
\right]

```

### 8.3.10 `\overset`, `\underset`, and `\sideset`

LaTeX provides `\stackrel` for placing a superscript above a binary relation. `amsmath` introduces somewhat more general commands, `\overset` and `\underset`. These can be used to place one symbol above or below another symbol, independently of whether it is a relation or something else. The input `\overset{*}{X}` will place a superscript-size `*` above the `X`; `\underset` performs the parallel operation that one would expect.

$$\overset{*}{X} \quad \underset{*}{X} \quad \overset{a}{\underset{b}{X}}$$

```

\left[ \overset{*}{X} \quad \underset{*}{X} \quad \overset{a}{\underset{b}{X}}
\right]

```

There is also a command called `\sideset` that serves a rather special purpose: it puts symbols in the subscript and superscript positions of large operator symbols such as  $\sum$  and  $\prod$ . A prime example is the case when you want to put a prime on a sum symbol. If there are no limits above or below the sum, you could just use `\nolimits`:

$$\sum' E_n. \tag{8.9}$$

```

\begin{equation}
\sum\nolimits' E_n.
\end{equation}

```

But if you want not only the prime but also limits on the sum symbol, things are not so easy. Suppose you want to add a prime on the sum symbol in an expression, like

$$\sum_{n < k, n \text{ odd}} n E_n \tag{8.10}$$

```

\begin{equation}
\sum_{n < k, \; n \; \mathrm{odd}} n E_n
\end{equation}

```

then you can use `\sideset` like this: `\sideset{}{'}\sum_{\dots}nE_n`. The extra pair of empty braces is explained by the fact that `\sideset` has the capability of putting an extra symbol or symbols at each corner of a large operator.

$$\prod_k^2 \prod_3^4 \sum'_{0 \leq i \leq m} E_i \beta x$$

```
\[
\sideset{}{'}\sum_{\le i \le m} E_i \beta x
\]
```

### 8.3.11 The `\smash` Command

The plain `TeX` command `\smash` retains the contents of a box but annihilates its height and depth. The `amsmath` package introduces the optional arguments `t` and `b` with the `\smash` command. `\smash[t]{...}` annihilates only the top of the box contents, retaining the bottom part, while `\smash[b]{...}` annihilates the bottom part and keeps the top.

$$X_j = (1/\sqrt{\lambda_j})X'_j \quad X_j = (1/\sqrt{\lambda_j})X'_j$$

```
\[ X_j=(1/\sqrt{\smash[b]{\lambda_j}})X_j'
\quad
X_j=(1/\sqrt{\lambda_j})X_j' \]
```

The previous example shows how the `\smash` command was used to limit the depth of the radical, which otherwise extends to encompass the depth of the subscript (right-hand formula in the above example).

### 8.3.12 The `\text` Command

The main use of the `\text` command, which is also available separately with the `amstext` package, is for words or phrases in a display. It is similar to the `LaTeX` command `\mbox` in its effects, but has a couple of advantages. If you would like a word or phrase of text in a subscript, you can type

```
..._{\text{word or phrase}}
```

which, apart from having a more descriptive name, is also slightly easier to enter than the equivalent `\mbox`, since the correct size is automatically chosen:

```
..._{\mbox{\scriptsize word or phrase}}
```

$$\mathbf{y} = \mathbf{y}' \quad \text{if and only if} \quad y'_k = \delta_k y_{\tau(k)}$$

```
\[ \mathbf{y}=\mathbf{y}' \quad \text{if and only if}
\text{if and only if}
y'_k=\delta_k y_{\tau(k)} \]
```

### 8.3.13 Operator Names

Math functions such as  $\log$ ,  $\sin$ , and  $\lim$  are traditionally set in roman type to help avoid confusion with single math variables, set in math italic. The more common ones have predefined names: `\log`, `\sin`, `\lim`, and so forth (see table 8.8 on page 226). New ones, however, come up all the time in mathematical papers. The `amsmath` package provides `\DeclareMathOperator` and `\DeclareMathOperator*` for producing new function names that will have the same typographical treatment. For instance, `\DeclareMathOperator{\xxx}{xxx}` produces  $\text{xxx}$  in the proper font and automatically adds proper spacing on either side when necessary, so that you get  $A\text{xxx}B$  instead of  $A\text{xxx}B$ . Examples of definitions of operator names are shown below (the `\`, in the definition of `\esssup` adds some space; see table 8.21 on page 252):

Input text	
<code>\DeclareMathOperator*\{esssup\}{ess\,sup}</code>	
<code>\DeclareMathOperator{\meas}{meas}</code>	
<code>\newcommand{\abs}[1]{\lvert#1\rvert}</code>	
<code>\newcommand{\norm}[1]{\lVert#1\rVert}</code>	
<code>\begin{align*}</code>	
<code>\norm{f}_\infty</code>	$\ f\ _\infty$
<code>\esssup_{x \in R^n} \abs{f(x)}</code>	$\operatorname{ess\,sup}_{x \in R^n}  f(x) $
<code>\meas_1\{u \in R_+^1 \colon f^*(u) &gt; \alpha\}</code>	$\operatorname{meas}_1\{u \in R_+^1 \colon f^*(u) > \alpha\}$
<code>\meas_n\{x \in R^n \colon \abs{f(x)} \geq \alpha\} \quad \forall \alpha &gt; 0.</code>	$\operatorname{meas}_n\{x \in R^n \colon  f(x)  \geq \alpha\} \quad \forall \alpha > 0.$
<code>\end{align*}</code>	

$$\|f\|_\infty = \operatorname{ess\,sup}_{x \in R^n} |f(x)|$$

$$\operatorname{meas}_1\{u \in R_+^1 : f^*(u) > \alpha\} = \operatorname{meas}_n\{x \in R^n : |f(x)| \geq \alpha\} \quad \forall \alpha > 0.$$

Output text
-------------

The starred form `\DeclareMathOperator*` is like `\DeclareMathOperator`; the only difference is the placement of subscripts and superscripts, as seen in the example above. In order to make the use of the vertical bar notation more flexible, `amsmath` defines the new commands `\lvert`, `\rvert`, `\lVert`, and `\rVert`, which are comparable to L<sup>A</sup>T<sub>E</sub>X's `\langle` and `\rangle`.

With `amsmath` the following operators are predefined: `\varlimsup`, `\varliminf`, `\varinjlim`, and `\varprojlim`. Here's what they look like in use:

$\overline{\lim}_{n \rightarrow \infty} Q(u_n, u_n - u^\#) \leq 0$	(8.11)	<code>\begin{gather}</code>	
$\underline{\lim}_{n \rightarrow \infty}  a_{n+1}  /  a_n  = 0$	(8.12)	<code>\varlimsup_{n \rightarrow \infty}</code>	<code>\mathcal{Q}(u_n, u_n - u^\{\#\}) \le 0</code>
$\overline{\lim}(m_i^\lambda)^* \le 0$	(8.13)	<code>\varliminf_{n \rightarrow \infty}</code>	<code>\left a_{n+1}\right /\left a_n\right  = 0</code>
$\varinjlim_{p \in S(A)} A_p \le 0$	(8.14)	<code>\varinjlim</code>	<code>(m_i^\lambda \cdot)^* \le 0</code>
		<code>\varprojlim</code>	<code>{p \in S(A)} A_p \le 0</code>
		<code>\end{gather}</code>	

### 8.3.14 `\mod` and Its Relatives

Commands `\mod`, `\bmod`, `\pmod`, and `\pod` are provided by the `amsofn` package to deal with the rather special spacing conventions of “mod” notation. `\bmod` and `\pmod` are available in L<sup>A</sup>T<sub>E</sub>X, but with `amsofn` the spacing of `\pmod` will adjust to a smaller value if it is used in a nondisplay formula. `\mod` and `\pod` are variants of `\pmod` preferred by some authors; `\mod` omits the parentheses, whereas `\pod` omits the “mod” and retains the parentheses.

$\gcd(k, l \bmod k)$	(8.15)	<code>\begin{equation}</code>	
		<code>\gcd</code>	<code>(k , l \bmod k)</code>
		<code>\end{equation}</code>	
$u \equiv v + 1 \pmod{n^2}$	(8.16)	<code>\begin{align}</code>	
$u \equiv v + 1 \bmod n^2$	(8.17)	<code>u &amp; \equiv v + 1 \pmod{n^2}</code>	<code>\\</code>
$u \equiv v + 1 (n^2)$	(8.18)	<code>u &amp; \equiv v + 1 \bmod{n^2}</code>	<code>\\</code>
		<code>u &amp; \equiv v + 1 \pod{n^2}</code>	
		<code>\end{align}</code>	

### 8.3.15 Fractions and Related Constructions

In addition to `\frac` (available in L<sup>A</sup>T<sub>E</sub>X), `amsmath` provides `\dffrac` and `\tfrac` as convenient abbreviations for `{\displaystyle\frac ... }` and `{\textstyle\frac ... }`.

$\frac{1}{k} \log_2 c(f)$		<code>\[ \frac{1}{k} \log_2 c(f) \quad</code>	
$\frac{1}{k} \log_2 c(f)$		<code>\tfrac{1}{k} \log_2 c(f) \quad \]</code>	
and $\sqrt{\frac{1}{k} \log_2 c(f)}$		<code>and</code>	
$\sqrt{\frac{1}{k} \log_2 c(f)}$		<code>\\$ \sqrt{\frac{1}{k} \log_2 c(f)} \quad</code>	
		<code>\sqrt{\dffrac{1}{k} \log_2 c(f)} \quad \\$.</code>	

For binomial expressions such as  $\binom{n}{k}$  the `amsmath` packages defines the commands `\binom`, `\dbinom`, and `\tbinom`.

$$\binom{k}{1}2^{k-1} + \binom{k}{2}2^{k-2} \quad (8.19)$$

and  $\binom{k}{1}2^{k-1} + \binom{k}{2}2^{k-2}$ .

```
\begin{equation}
\binom{k}{1}2^{k-1}+\tbinom{k}{2}2^{k-2}
\end{equation}
and
$\binom{k}{1}2^{k-1}+\dbinom{k}{2}2^{k-2}$.
```

`\binom`, and its variants `\dbinom` and `\tbinom`, as well as `\frac` and its variants `\dfrac` and `\tfrac` are implemented using the generalized fraction command `\genfrac`, which has six parameters.

```
\genfrac{ldelim}{rdelim}{thick}{style}{num}{denom}
```

The first two parameters *ldelim* and *rdelim* are the left and right delimiters, respectively. The third parameter *thick* allows you to override the line thickness (for instance `\binom` uses this to set the line thickness to zero, i.e., invisible). If this argument is left empty, the line thickness defaults to “normal”. The fourth parameter is the mathematics style override. It can take integer values in the range 0–3 to select, respectively, `\displaystyle`, `\textstyle`, `\scriptstyle`, and `\scriptscriptstyle`. Finally, the fifth argument *num* is the numerator, while the sixth *denom* is the denominator of the fraction.

To illustrate, here is how `\frac`, `\tfrac`, and `\binom` might be defined.

```
\newcommand{\frac}[2]{\genfrac{}{}{}{}{#1}{#2}}
\newcommand{\tfrac}[2]{\genfrac{}{}{1}{#1}{#2}}
\newcommand{\binom}[2]{\genfrac{()}{()}{0pt}{}{#1}{#2}}
```

Other examples are the following re-implementation of TeX’s fraction primitives.

```
\renewcommand{\over}[2]{%
\genfrac{}{}{}{#1}{#2}}
\renewcommand{\overwithdelims}[2]{%
\genfrac{\langle\rangle}{\langle\rangle}{}{#1}{#2}}
\[\over{n+1}{n}\quad\overwithdelims{n+1}{n}\ \]
\renewcommand{\atop}[2]{%
\genfrac{}{}{0pt}{}{#1}{#2}}
\renewcommand{\atopwithdelims}[2]{%
\genfrac{()}{()}{0pt}{}{#1}{#2}}
\[\atop{n+2}{n}\quad\atopwithdelims{n+2}{n}\ \]
\renewcommand{\above}[2]{%
\genfrac{}{}{1pt}{}{#1}{#2}}
\renewcommand{\abovewithdelims}[2]{%
\genfrac{[]{}{1pt}{}{#1}{#2}}
\[\above{n-3}{n}\quad\abovewithdelims{n-3}{n}\ \]
```

Of course, if you want to use a particular notation implemented with `\genfrac` repeatedly throughout your document you will do yourself (and your publisher) a favor if you define a meaningful command name with `\newcommand` as an abbreviation for that notation, as in the examples above.

### 8.3.16 Continued Fractions

A continued fraction can be obtained as follows:

$$\frac{1}{\sqrt{2} + \frac{1}{\sqrt{3} + \frac{1}{\sqrt{4} + \frac{1}{\sqrt{5} + \frac{1}{\sqrt{6} + \dots}}}}}$$

(8.20)

```

\begin{equation}
\cfrac{1}{\sqrt{2}+
\cfrac{1}{\sqrt{3}+
\cfrac{1}{\sqrt{4}+
\cfrac{1}{\sqrt{5}+
\cfrac[r]{1}{\sqrt{6}+\dotsb
}}}}}
\end{equation}
    
```

Left or right positioning of any of the numerators is achieved by using the optional argument `[l]` or `[r]` with the `\cfrac` command.

### 8.3.17 Big-g-g-g Delimiters

In order to better control the sizes of math delimiters, basic TeX introduces four commands `\big`, `\Big`, `\bigg` and `\Bigg`, which produce ever larger versions of the delimiter specified as parameter. These commands can be used with any of the delimiters that can follow the `\left` or `\right` command (see tables 8.9, 8.10, and 8.13 on page 227). Moreover, for each of the four commands above, three variants exist for use as an opening symbol (e.g., `\bigl`), as a binary relation (e.g., `\Bigm`), or as a closing symbol (e.g., `\Biggr`).<sup>4</sup> Whereas, with basic TeX, the sizes of these delimiters are fixed, with the `amsmath` package the sizes adapt to the size of the surrounding material.

$$\left( \mathbf{E}_y \int_0^{t_\varepsilon} L_{x,y^x(s)} \varphi(x) ds \right)$$

```

\left[
\biggl(\mathbf{E}_{-y}\int_0^{t_{\varepsilon}}
L_{x,y^x(s)}\varphi(x)\,ds \biggr)
\right]
\left[
\Bigl(\mathbf{E}_{-y}\int_0^{t_{\varepsilon}}
L_{x,y^x(s)}\varphi(x)\,ds \Bigl)
\right]
    
```

<sup>4</sup> See table 7.13 on page 213 for a discussion of the various math symbol types.

## 8.4 Matrix-Like Environments and Commutative Diagrams

### 8.4.1 The cases Environment

“Case” constructions can be produced using the `cases` environment.

$$P_{r-j} = \begin{cases} 0 & \text{if } r-j \text{ is odd,} \\ r!(-1)^{(r-j)/2} & \text{if } r-j \text{ is even.} \end{cases} \quad (8.21)$$

```

\begin{equation}
P_{\{r-j\}}=
\begin{cases}
0& \text{\text{if } $r-j$ is odd},\\
r!\,(-1)^{\{(r-j)/2\}}& \text{\text{if } $r-j$ is even.}
\end{cases}
\end{equation}

```

Notice the use of `\text` and the embedded math.

### 8.4.2 The Matrix Environments

The matrix environments are similar to L<sup>A</sup>T<sub>E</sub>X’s `array`, except they do not have an argument specifying the format of the columns. Instead, a default format is provided: up to 10 centered columns. The examples below show how to use the matrix environments `matrix`, `pmatrix`, `bmatrix`, `vmatrix`, and `Vmatrix`:

$$\begin{matrix} 0 & 1 & \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} & \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \\ 1 & 0 & \begin{vmatrix} a & b \\ c & d \end{vmatrix} & \left\| \begin{matrix} 1 & 0 \\ 0 & 1 \end{matrix} \right\| \end{matrix}$$

```

\begin{gather*}
\begin{matrix} 0 & 1 & \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} & \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \\ 1 & 0 & \begin{vmatrix} a & b \\ c & d \end{vmatrix} & \left\| \begin{matrix} 1 & 0 \\ 0 & 1 \end{matrix} \right\| \end{matrix}
\end{gather*}

```

The maximum number of columns is determined by the counter `MaxMatrixCols`, which you can change using L<sup>A</sup>T<sub>E</sub>X’s standard counter commands. For example, suppose you have a large matrix with 19 or 20 columns, then you can do something like this:

```

\begin{equation}
\setcounter{MaxMatrixCols}{20}
A=\begin{pmatrix}
...&...&...&...&...&...&...&...&...&...&...&...&...&...&...&...&...&...&...&...&...
... \\
...
\end{pmatrix}
\end{equation}
\setcounter{MaxMatrixCols}{10}

```

As counters are global in L<sup>A</sup>T<sub>E</sub>X, you might want to reset the value of `MaxMatrixCols` to its default value of 10 after finishing your wide matrix, since with a high value, L<sup>A</sup>T<sub>E</sub>X must work a lot harder to typeset a matrix.

To produce a small matrix suitable for use in text, use the `smallmatrix` environment.

To show the effect of the matrix on surrounding lines inside a paragraph, we put it here:  $\begin{pmatrix} a & b \\ c & d \end{pmatrix}$  and follow it with enough text to ensure that there is at least one full line below the matrix.

To show the effect of the matrix on surrounding lines inside a paragraph, we put it here:

```
\begin{math}
\left( \begin{smallmatrix}
a&b\ c&d
\end{smallmatrix} \right)
\end{math}
```

and follow it with enough text to ensure that there is at least one full line below the matrix.

A row of dots in a matrix, spanning a given number of columns, can be obtained with the command:

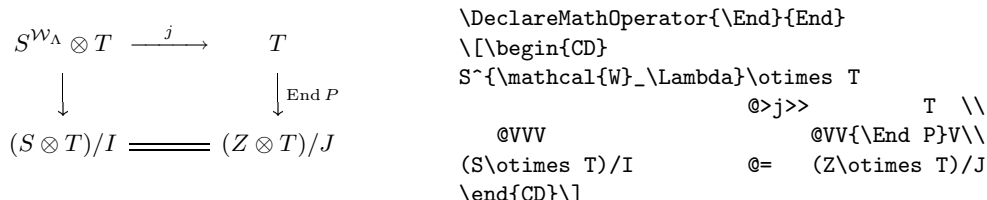
```
\hdotsfor[spacing-factor]{number}
```

The spacing of the dots can be varied by using the optional parameter *spacing-factor*, for example, `\hdotsfor[1.5]{3}`. The number in square brackets multiplies the spacing between the dots; the normal value is one.

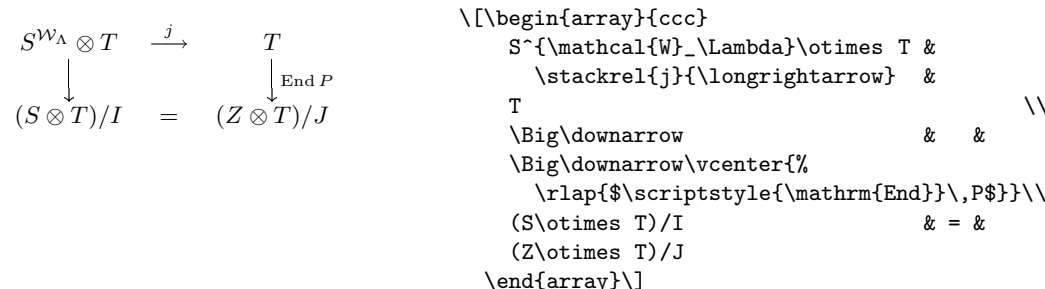


### 8.4.4 Commutative Diagrams

The commutative diagram commands of  $\mathcal{A}\mathcal{M}\mathcal{S}$ - $\text{\TeX}$  are not included in the `amsmath` package, but are available as a separate package, `amscd`. This conserves memory for users who do not need commutative diagrams. The `picture` environment can be used for complex commutative diagrams, but for simple diagrams without diagonal arrows the `amscd` commands are more convenient.<sup>5</sup>



A similar result, which does not look quite as good, can be produced in ordinary  $\text{\LaTeX}$  by:



When using the `amscd` package, you will obtain longer horizontal arrows and improved spacing between elements of the diagram.

In the `CD` environment the commands `@>>>`, `@<<<`, `@VVV`, and `@AAA` give (respectively) right, left, down, and up arrows. For people with keyboards lacking the angle brackets the notations `@))` and `@((` are available as alternatives.

For the horizontal arrows, material between the first and second `>` or `<` symbols will be typeset as a superscript, and material between the second and third will be typeset as a subscript. Similarly, material between the first and second, or second and third, `A`'s or `V`'s of vertical arrows will be typeset as left or right "sidescripts." This was used in the first example above to place the operator "End *P*" to the right of the arrow.

The final example again shows the use of `\DeclareMathOperator`.

<sup>5</sup> Much more extensive commutative diagram packages are Kristoffer Rose's `XY-pic` system [?], Paul Taylor's `Commutative Diagram` package [?], and the `Diagram 3` system by Francis Borceux [?].

$$\begin{array}{ccccc}
 \text{cov}(\mathcal{L}) & \longrightarrow & \text{non}(\mathcal{K}) & \longrightarrow & \text{cf}(\mathcal{K}) \\
 \downarrow & & \uparrow & & \uparrow \\
 \text{add}(\mathcal{L}) & \longrightarrow & \text{add}(\mathcal{K}) & \longrightarrow & \text{cov}(\mathcal{K})
 \end{array}$$

```

\begin{equation*}
\DeclareMathOperator{\add}{add}
\DeclareMathOperator{\cf}{cf}
\DeclareMathOperator{\cov}{cov}
\DeclareMathOperator{\non}{non}
\begin{CD}
\cov(\mathcal{L}) @>>> \non(\mathcal{K}) \\
@VVV @AAA @AAA @>>> \cf(\mathcal{K}) \\
\add(\mathcal{L}) @>>> \add(\mathcal{K}) @>>> \cov(\mathcal{K})
\end{CD}
\end{equation*}

```

## 8.5 Alignment Structures for Equations

The `amsmath` package defines several environments for creating multiline display equations. They perform similarly to L<sup>A</sup>T<sub>E</sub>X's `equation` and `eqnarray` environments. The following structures are discussed in the next sections.

<code>align</code>	<code>align*</code>	alignment at a single place
<code>flalign</code>	<code>flalign*</code>	spaced-out variants of the above
<code>alignat</code>	<code>alignat*</code>	alignment with space control
<code>equation</code>	<code>equation*</code>	one-line formula
<code>gather</code>	<code>gather*</code>	combining formula without alignment
<code>multline</code>	<code>multline*</code>	multiline equation (one equation number)
<code>split</code>		splitting long formulas

Some of these multiline display environments allow you to align parts of the formula. In contrast to the original L<sup>A</sup>T<sub>E</sub>X environments `eqnarray` and `eqnarray*`, the structures implemented by the `amsmath` package use a different concept for marking the alignment points: while `eqnarray` is similar to an `array` environment with a `{rcl}` preamble and therefore uses two ampersand characters surrounding the part that should be aligned, in the `amsmath` structures you should mark the alignment point (or points in `alignat`, for example) only with a single ampersand character, placing it to the left of the character that should be aligned with previous or following lines.

The `amsmath` structures give correct spacing around the alignment points, while the `eqnarray` environment produces extra spaces depending on the parameter settings for `array`. The difference can be seen clearly in the next example, where we typeset the same equation using the `equation`, `align`, and `eqnarray` environments; ideally all three should produce the same result, but the `eqnarray` environment comes out too wide.



number of “align” structures. For an argument of  $n$ , the number of ampersand characters per line is  $2n - 1$  (one ampersand for alignment within each align structure, and ampersands to separate the align structures from one another).

The special environment `flalign` is a form of the `align` environment with added space between the component align structures.

$L_1 = R_1$	$L_2 = R_2$	(8.33)	<code>\begin{align}</code>
			<code>L_1 &amp; = R_1 &amp; \quad L_2 &amp; = R_2 \quad \backslash\backslash</code>
$L_3 = R_3$	$L_4 = R_4$	(8.34)	<code>L_3 &amp; = R_3 &amp; \quad L_4 &amp; = R_4</code>
			<code>\end{align}</code>
$L_1 = R_1$	$L_2 = R_2$	(8.35)	<code>\begin{alignat}{2}</code>
			<code>L_1 &amp; = R_1 &amp; \quad L_2 &amp; = R_2 \quad \backslash\backslash</code>
$L_3 = R_3$	$L_4 = R_4$	(8.36)	<code>L_3 &amp; = R_3 &amp; \quad L_4 &amp; = R_4</code>
			<code>\end{alignat}</code>
$L_1 = R_1$	$L_2 = R_2$	(8.37)	<code>\begin{flalign}</code>
			<code>L_1 &amp; = R_1 &amp; \quad L_2 &amp; = R_2 \quad \backslash\backslash</code>
$L_3 = R_3$	$L_4 = R_4$	(8.38)	<code>L_3 &amp; = R_3 &amp; \quad L_4 &amp; = R_4</code>
			<code>\end{flalign}</code>
$L_1 = R_1$	$L_2 = R_2$		<code>\begin{flalign*}</code>
			<code>L_1 &amp; = R_1 &amp; \quad L_2 &amp; = R_2 \quad \backslash\backslash</code>
$L_3 = R_3$	$L_4 = R_4$		<code>L_3 &amp; = R_3 &amp; \quad L_4 &amp; = R_4</code>
			<code>\end{flalign*}</code>

More examples are shown in section 8.7.6 on page 261.

### 8.5.3 Split Equations without Alignment

The `multline` environment is a variation of the `equation` environment used for equations that do not fit on a single line. The first line of a `multline` will be at the left margin and the last line at the right margin except for an indentation on both sides whose amount is equal to `\multlinegap`. The value of `\multlinegap` can be changed using L<sup>A</sup>T<sub>E</sub>X’s `\setlength` and `\addtolength` commands. If `multline` contains more than two lines, any lines other than the first and last will be centered individually within the display width (unless option `fleqn` is in effect). It is, however, possible to force a line to the left or the right with the `\shoveleft` and `\shoveright` commands.

	<code>\begin{multline}</code>	
First line of equation	<code>\text{First line of equation}</code>	<code>\\</code>
Centered Middle line	<code>\text{Centered Middle line}</code>	<code>\\</code>
Right Middle line	<code>\shoveright{\text{Right Middle line}}</code>	<code>\\</code>
Other centered Middle	<code>\shoveleft{\text{Left Middle line}}</code>	<code>\\</code>
Left Middle line	<code>\text{Last line of equation}</code>	<code>\\</code>
Last line of equation	<code>\end{multline}</code>	

(8.39)

More examples are shown in section 8.7.2 on page 258.

### 8.5.4 Split Equations with Alignment

Like `multline`, the `split` environment is for single equations that are too long to fit on a single line and hence must be split into multiple lines. Unlike `multline`, however, the `split` environment provides for alignment among the split lines, using an ampersand to mark alignment points, as usual. In addition (unlike the other `amsmath` equation structures) the `split` environment provides no numbering because it is intended to be used only inside some other displayed equation structure, such as `equation`, `align`, or `gather`. These outer environments will provide the numbering.

$(a + b)^4 = (a + b)^2(a + b)^2$	<code>\begin{equation}</code>
$= (a^2 + 2ab + b^2)(a^2 + 2ab + b^2)$	<code>\begin{split}</code>
$= a^4 + 4a^3b + 6a^2b^2 + 4ab^3 + b^4$	<code>(a+b)^4 &amp;= (a+b)^2 (a+b)^2</code>
	<code>&amp;= (a^2+2ab+b^2)(a^2+2ab+b^2)</code>
	<code>&amp;= a^4+4a^3b+6a^2b^2+4ab^3+b^4</code>
	<code>\end{split}</code>
	<code>\end{equation}</code>

(8.40)

When the `tbtags` option is specified, the equation number for the `split` environment will be put on the last (resp. first) line if the equation number is on the right (resp. left). By default, the `centertags` option is in effect, putting the equation number centered vertically on the height of the `split`, provided there is enough room for it.

$(a + b)^3 = (a + b)(a + b)^2$	<code>\begin{equation}</code>
$= (a + b)(a^2 + 2ab + b^2)$	<code>\begin{split}</code>
$= a^3 + 3a^2b + 3ab^2 + b^3$	<code>(a+b)^3 &amp;= (a+b) (a+b)^2</code>
	<code>&amp;= (a+b)(a^2+2ab+b^2)</code>
	<code>&amp;= a^3+3a^2b+3ab^2+b^3</code>
	<code>\end{split}</code>
	<code>\end{equation}</code>

(8.41)

More examples are shown in section 8.7.1 on page 255.

### 8.5.5 Alignment Environments as Parts of Displays

In addition to the `split` environment, there are some other equation alignment environments that do not constitute an entire display. They are self-contained units that can be used inside other formulae, or set side by side. The environment names are: `aligned`, `gathered`, and `alignedat`. These environments take an optional argument to specify their vertical positioning with respect to the material on either side. The default alignment is centered (`[c]`), and its effect is seen in the following example.

$x^2 + y^2 = 1$	$(a + b)^2 = a^2 + 2ab + b^2$	$(a + b) \cdot (a - b) = a^2 - b^2$	<pre>\begin{equation*} \begin{aligned} x^2 + y^2 &amp;= 1 \\ x &amp;= \sqrt{1-y^2} \end{aligned} \end{equation*}</pre>	<pre>\begin{aligned} (a + b)^2 &amp;= a^2 + 2ab + b^2 \\ (a + b) \cdot (a - b) &amp;= a^2 - b^2 \end{aligned}</pre>	<pre>\begin{equation*} \begin{aligned} x^2 + y^2 &amp;= 1 \\ x &amp;= \sqrt{1-y^2} \end{aligned} \end{equation*}</pre>	<pre>\quad</pre>
-----------------	-------------------------------	-------------------------------------	--	---	--	------------------

The same mathematics can now be typeset using different vertical alignments for the environments.

$x^2 + y^2 = 1$	$(a + b)^2 = a^2 + 2ab + b^2$	$(a + b) \cdot (a - b) = a^2 - b^2$	<pre>\begin{equation*} \begin{aligned}[b] x^2 + y^2 &amp;= 1 \\ x &amp;= \sqrt{1-y^2} \end{aligned} \end{equation*}</pre>	<pre>\begin{aligned}[t] (a + b)^2 &amp;= a^2 + 2ab + b^2 \\ (a + b) \cdot (a - b) &amp;= a^2 - b^2 \end{aligned}</pre>	<pre>\begin{equation*} \begin{aligned}[b] x^2 + y^2 &amp;= 1 \\ x &amp;= \sqrt{1-y^2} \end{aligned} \end{equation*}</pre>	<pre>\quad</pre>
-----------------	-------------------------------	-------------------------------------	---	--	---	------------------

### 8.5.6 Vertical Spacing and Page Breaks in Equation Structures

You can use the `\[dimension]` command to get extra vertical space between lines in all the `amsmath` displayed equation environments, as is usual in `LATEX`. Unlike `eqnarray`, the `amsmath` environments do not allow page breaks between lines, unless `\displaybreak` or `\allowdisplaybreaks` is used. The reason

for this is that page breaks in such situations should receive individual attention from the author. `\displaybreak` must go before the `\\` where it is supposed to take effect. Like L<sup>A</sup>T<sub>E</sub>X's `\pagebreak`, `\displaybreak` takes an optional argument between zero and four denoting the desirability of the page break. `\displaybreak[0]` means "it is permissible to break here" without encouraging a break; `\displaybreak` with no optional argument is the same as `\displaybreak[4]` and forces a break.

There is also an optional argument for `\allowdisplaybreaks`. This command obeys the usual L<sup>A</sup>T<sub>E</sub>X scoping rules. The normal way of limiting its scope is to put `{\allowdisplaybreaks` at the beginning and `}` at the end of the desired range. Within the scope of an `\allowdisplaybreaks` command, the `\\*` command can be used to prohibit a page break, as usual.

### 8.5.7 The `\intertext` Command

The `\intertext` command is used for a short interjection of one or two lines of text in the middle of a display alignment. Its salient feature is the preservation of alignment, which would not be possible if you simply ended the display and then started it up again afterwards. `\intertext` may only appear immediately after a `\\` or `\\*` command.

$A_1 = N_0(\lambda; \Omega') - \phi(\lambda; \Omega'), \quad (8.42)$	(8.42)	<pre>\begin{align} A_1&amp;=N_0(\lambda;\Omega') - &amp;\phi(\lambda;\Omega'), \\ A_2&amp;=\phi(\lambda;\Omega')\phi(\lambda;\Omega), \\ &amp;\phi(\lambda;\Omega), \\ &amp;\phi(\lambda;\Omega),</pre>
and finally		<pre>\intertext{and finally} A_3&amp;=\mathcal{N}(\lambda;\omega).</pre>
$A_2 = \phi(\lambda; \Omega')\phi(\lambda; \Omega), \quad (8.43)$	(8.43)	<pre>\end{align}</pre>
$A_3 = \mathcal{N}(\lambda; \omega). \quad (8.44)$	(8.44)	<pre>\end{align}</pre>

Here the words "and finally" fall outside the display at the left margin.

## 8.6 Miscellaneous

This section discusses `amsmath` commands that have not been introduced yet, and it gives a list of the document class files that come with the  $\mathcal{A}\mathcal{M}\mathcal{S}$ -L<sup>A</sup>T<sub>E</sub>X distribution.

### 8.6.1 Equation Numbers

Each environment, except for `split`, has both starred and unstarred forms, where the unstarred forms have automatic numbering, using L<sup>A</sup>T<sub>E</sub>X's `equation`

counter. The number on any particular line can be suppressed by putting `\notag` before the `\`. You can also override it with a tag of your own design using

<code>\tag{label}</code>	<code>\tag*{label}</code>
--------------------------	---------------------------

where *label* can be any arbitrary text to be used to number the equation.

The starred form, `\tag*`, causes the *label* to be typeset without any annotations like parentheses that might otherwise be added by the document class. `\tag` and `\tag*` can also be used in the starred versions of all the `amsmath` alignment environments.

$x^2 + y^2 = z^2$	(8.45)	<code>\begin{gather}</code>	<code>x^2+y^2 = z^2 \label{eq:r2}</code>	<code>\</code>
$x^3 + y^3 = z^3$			<code>x^3+y^3 = z^3 \notag</code>	<code>\</code>
$x^4 + y^4 = r^4$	(*)		<code>x^4+y^4 = r^4 \tag{*\$}\$}</code>	<code>\</code>
$x^5 + y^5 = r^5$	*		<code>x^5+y^5 = r^5 \tag*{*\$}\$}</code>	<code>\</code>
$x^6 + y^6 = r^6$	(8.45')	<code>\end{gather}</code>	<code>x^6+y^6 = r^6 \tag{\ref{eq:r2}\$'\$}</code>	

Notice the use of the `\label` and `\ref` commands in the previous example to allow subnumbering of equations.

When `leqno` is specified as an option to the `amsmath` package, the equation number will be printed at the left side of the equation (by default, with `amsmath`, it comes out at the right).

(8.46)	$\sin^2 \eta + \cos^2 \eta = 1$	<code>\begin{equation}</code>	<code>\sin^2\eta + \cos^2\eta = 1</code>	<code>\end{equation}</code>
--------	---------------------------------	-------------------------------	--	-----------------------------

## 8.6.2 Resetting the Equation Counter

In  $\text{\LaTeX}$ , if you want to have equations numbered within sections—that is, have equation numbers (1.1), (1.2), . . . , (2.1), (2.2), . . . , in sections 1, 2, and so forth—you would probably redefine `\theequation`:

```
\renewcommand{\theequation}{\thesection.\arabic{equation}}
```

But now you have to reset the equation number by hand at the beginning of each new section or chapter. To make this a little more convenient, `amsmath` provides a command `\numberwithin`. To have equation numbering tied to section numbering, with automatic reset of the equation counter, the command is

```
\numberwithin{equation}{section}
```

As the name implies, `\numberwithin` can be applied to other counters besides the equation counter, but the results may not be satisfactory in all cases because of potential complications. Normal L<sup>A</sup>T<sub>E</sub>X methods should be used where available, for example, in `\newtheorem`.<sup>6</sup>

To make cross-references to equations easier, an `\eqref` command is provided. This automatically supplies the parentheses around the equation number, and adds an italic correction before the closing parenthesis, if necessary. To refer to an equation that was labeled with the label `e:baset`, the usage would be `\eqref{e:baset}`.

### 8.6.3 Subordinate numbering sequences

The `amsmath` package provides also a `subequations` environment to make it easy to number equations in a particular group with a subordinate numbering scheme. For example

```
\begin{subequations}
...
\end{subequations}
```

causes all numbered equations within that part of the document to be numbered (4.9a) (4.9b) (4.9c) . . . , if the preceding numbered equation was (4.8). A `\label` command immediately following `\begin{subequations}` produces a `\ref` of the parent number 4.9, not 4.9a. The counters used by the `subequations` environment are `parentequation` and `equation`. They can be set by the L<sup>A</sup>T<sub>E</sub>X commands `\addtocounter`, `\setcounter`, `\value`, etc.. Moreover, the style of the subordinate numbers, are controlled using standard L<sup>A</sup>T<sub>E</sub>X methods (see Section A.1.3). For example, redefining `\theequation` as follows will produce roman numerals.

```
\begin{subequations}
\renewcommand{\theequation}{\theparentequation \roman{equation}}
...
\end{subequations}
```

### 8.6.4 Fine-Tuning Spacing in Math Mode

Although T<sub>E</sub>X generally does a good job of spacing elements of formulae inside mathematics, it is sometimes necessary to fine-tune the position of one or two of those elements. Therefore, the spacing commands shown in table 8.21 on the next page are provided. Both the spelled-out and abbreviated forms of these commands are robust, and they can also be used outside of math.

<sup>6</sup> See also the discussion of the `\addtoreset` command on page 23.

Positive space			Negative space		
Abb.	ex.	Spelled out	Abb.	ex.	Spelled out
\,	$x x$	\thinspace	\!	$x x$	\negthinspace
\:	$x x$	\medspace		$x x$	\negmedspace
\;	$x x$	\thickspace		$x x$	\negthickspace
	$x x$	\quad			
	$x x$	\qquad			

Table 8.21: The mathematical spacing commands

For allow you to further fine-tune the spacing in math expressions the command `\mspace` is defined. Its only argument is a L<sup>A</sup>T<sub>E</sub>X length expressed in ‘math units’. One math unit, or `\mu`, is equal to 1/18 em (see also table A.1 on page 476). Thus, to get a negative `\quad` you could write `\mspace{-18.0\mu}`.

### 8.6.5 A Few Points to Note

( $\mathcal{L}$  151–52)

Many of the commands added by the `amsmath` package are fragile and will need to be `\protected` in commands with “moving arguments.”

With the various alignment environments available in the `amsmath` package, the `eqnarray` environment is no longer needed. Furthermore, since it does not prevent overlapping of the equation numbers with wide formulae, as most of the `amsmath` alignments do, using the `amsmath` alignments seems better. `amsmath` reimplements the L<sup>A</sup>T<sub>E</sub>X `equation` environment as a one-line `gather` environment, and adds an unnumbered version, `equation*`, for symmetry. Note, however, that the command `\verb` might not work in the alignment environments.

`\nonumber` is interchangeable with `\notag`; the latter seems slightly preferable, for consistency with the name `\tag`.

### 8.6.6 Options and Sub-Packages to the `amsmath` Package

A few options are recognized by the `amsmath` package and the classes provided by  $\mathcal{A}\mathcal{M}\mathcal{S}$ -L<sup>A</sup>T<sub>E</sub>X.<sup>7</sup> They affect the positioning of math operator limits or `\tags`.

`centertags` (default) The text of the tag of a `split` environment is vertically centered with respect to its total height.

`tbtags` “Top-or-bottom tags”. The text of the tag of a `split` environment is placed level with the last (resp. first) line, if numbers are on the right (resp. left).

<sup>7</sup> This is only true for the L<sup>A</sup>T<sub>E</sub>X<sub>2 $\epsilon$</sub>  release of  $\mathcal{A}\mathcal{M}\mathcal{S}$ -L<sup>A</sup>T<sub>E</sub>X. Older versions of  $\mathcal{A}\mathcal{M}\mathcal{S}$ -L<sup>A</sup>T<sub>E</sub>X realize these options as sub-packages.

- `intlimits` Like `sumlimits`, but for integral symbols.
- `nointlimits` (default) Opposite of `intlimits`.
- `namelimits` (default) Like `sumlimits`, but for certain “operator names” such as `det`, `inf`, `lim`, `max`, `min`, that traditionally have subscripts placed underneath when they occur in a displayed equation.
- `nonamelimits` Opposite of `namelimits`.
- `sumlimits` (default) Place subscripts and superscripts of summation symbols above and below, in displayed equations. This option also affects other symbols of the same type— $\prod$ ,  $\coprod$ ,  $\otimes$ ,  $\oplus$ , and so forth—but excluding integrals (see `intlimits`).
- `nosumlimits` Place subscripts and superscripts of summation-type symbols to the side, even in displayed equations.

The following three options are usually global document options and are thus set on the `\documentclass` command. They are, however, also recognized when the `amsmath` package is loaded with the `\usepackage` command.

- `leqno` Place equation numbers on the left.
- `reqno` Place equation numbers on the right (default).
- `fleqn` Position equations at a fixed indent from the left margin rather than centered in the text column.

The  $\mathcal{A}\mathcal{M}\mathcal{S}$ - $\mathcal{L}\mathcal{A}\mathcal{T}\mathcal{E}\mathcal{X}$  distribution consists of a set of components, which can be loaded independently with the `\usepackage` command. The single most noteworthy package is probably `amsmath`, but the others can be used individually. Note that the `amsbsy`, `amsopn`, and `amstext` packages are included automatically when you use the `amsmath` package.

- `amsmath` Defines extra environments for multiline displayed equations, plus a number of other enhancements for math.
- `amsbsy` Defines the `\boldsymbol` and `\pmb` (poor man’s bold) commands.
- `amsopn` Provides `\DeclareMathOperator` for defining new “operator names” like `\sin` and `\lim`.
- `amstext` Provides a `\text` command for typesetting a fragment of text inside a display.

Other packages, providing supplementary functionality, should be loaded explicitly. Only parts of these packages are described in the present chapter.

They are mentioned here for completeness.

- `amscd` Defines some commands for easing the generation of commutative diagrams by introducing the `CD` environment (see Section 8.4.4). There is no support for diagonal arrows.
- `amsintx` Provides more descriptive command syntax for integrals and sums (not released yet).
- `amsthm` Provides a `proof` environment and extensions for the `\newtheorem` command.
- `amsxtra` Provides certain odds and ends such as `\fracwithdelims` and `\accentedsymbol` (see Section 8.3.4).
- `upref` Makes `\ref` print cross-reference numbers always in an upright/roman font regardless of context.

Finally, there are a few packages which come with the `AMSFonTS` distribution.

- `amsfonts` defines the `\mathfrak` and `\mathbb` commands and sets up the fonts `msam` (extra math symbols A), `msbm` (extra math symbols B, and blackboard bold), `eufm` (Euler Fraktur), extra sizes of `cmmib` (bold math italic and bold lowercase Greek), and `cmsy` (bold math symbols and bold script), for use in mathematics.
- `amssymb` defines the names of all the math symbols available with the `AMS` fonts collection. This package loads the `amsfonts` package.
- `eufrak` Set up the Fraktur letters.
- `eucal` Makes `\mathcal` use the Euler script instead of the usual Computer Modern script letters.

All these packages recognize the `psamsfonts` option, which will use the Y&Y/Blue Sky Research version of the `AMSFonTS` collection (which is free available on CTAN).

### 8.6.7 `AMS-LATEX` Document Classes

The `AMS-LATEX` package comes with a pair of document classes called `amsart` and `amsbook`, corresponding to `LATEX`'s `article` and `book`. They are primarily designed to prepare manuscripts for submission to the AMS, but there is nothing to prohibit their use for other purposes. With these class files the `amsmath` package is automatically included, so that you can start your document simply with `\documentclass{amsart}` or `\documentclass{amsbook}`.

## 8.7 Examples of Multiple-Line Equation Structures

On the following pages we show a lot of real-life examples of the alignment environments discussed earlier. The lines indicating the margins around the typeset examples are not part of the environments but have been added to make the marginal spacing stand out clearly.

### 8.7.1 The split Environment

The `split` environment is not an independent environment but should be used inside something else, such as `equation` or `align`.

If there is not enough room for it, the equation number for a `split` will be shifted to the previous line when equation numbers are on the left; the number shifts down to the next line when numbers are on the right.

When you do not want an equation number, use the `equation*` environment.

$$\begin{aligned}
 f_{h,\varepsilon}(x,y) &= \varepsilon \mathbf{E}_{x,y} \int_0^{t_\varepsilon} L_{x,y_\varepsilon(\varepsilon u)} \varphi(x) du \\
 &= h \int L_{x,z} \varphi(x) \rho_x(dz) \\
 &\quad + h \left[ \frac{1}{t_\varepsilon} \left( \mathbf{E}_y \int_0^{t_\varepsilon} L_{x,y^x(s)} \varphi(x) ds - t_\varepsilon \int L_{x,z} \varphi(x) \rho_x(dz) \right) \right. \\
 &\quad \left. + \frac{1}{t_\varepsilon} \left( \mathbf{E}_y \int_0^{t_\varepsilon} L_{x,y^x(s)} \varphi(x) ds - \mathbf{E}_{x,y} \int_0^{t_\varepsilon} L_{x,y_\varepsilon(\varepsilon s)} \varphi(x) ds \right) \right] \tag{8.47}
 \end{aligned}$$

This was produced by the following input (the `TeX` command `\phantom` is used to leave a space equal to the width of its argument):

```

\begin{equation}
\begin{split}
f_{h,\varepsilon}(x,y)
&= \varepsilon \mathbf{E}_{x,y} \int_0^{t_\varepsilon} L_{x,y_\varepsilon(\varepsilon u)} \varphi(x) du \\
&= h \int L_{x,z} \varphi(x) \rho_x(dz) \\
&\quad + h \biggl[ \frac{1}{t_\varepsilon} \biggl( \mathbf{E}_y \int_0^{t_\varepsilon} L_{x,y^x(s)} \varphi(x) \, ds
-t_\varepsilon \int L_{x,z} \varphi(x) \rho_x(dz) \biggr) \\
&\quad + \frac{1}{t_\varepsilon} \biggl( \mathbf{E}_y \int_0^{t_\varepsilon} L_{x,y^x(s)} \varphi(x) \, ds
- \mathbf{E}_{x,y} \int_0^{t_\varepsilon} L_{x,y_\varepsilon(\varepsilon s)} \varphi(x) \, ds \biggr) \biggr]
\end{split}
\end{equation}

```

```
\end{split}
\end{equation}
```

If the option `centertags` is included in the options list of the `amsmath` package, the equation numbers for `split` environments will be centered vertically on the height of the `split`, as shown in the example below.

$$\left| \begin{aligned} |I_2| &= \left| \int_0^T \psi(t) \left\{ u(a, t) - \int_{\gamma(t)}^a \frac{d\theta}{k(\theta, t)} \int_a^\theta c(\xi) u_t(\xi, t) d\xi \right\} dt \right| \\ &\leq C_6 \left\| f \int_\Omega \left| \tilde{S}_{a,-}^{-1,0} W_2(\Omega, \Gamma_l) \right| \left\| |u| \overset{\circ}{\rightarrow} W_2^{\tilde{A}}(\Omega; \Gamma_r, T) \right\| \right\|. \end{aligned} \right. \quad (8.48)$$

This is produced by the following input:

```
\begin{equation}
\begin{split}
|I_2|
&= \left| \int_0^T \psi(t) \right.
&\quad \left. \left\{ u(a, t) - \int_{\gamma(t)}^a \frac{d\theta}{k(\theta, t)} \right. \right.
&\quad \left. \left. \int_a^\theta c(\xi) u_t(\xi, t) d\xi \right\} dt \right.
&\quad \left. \right| \\
&\leq C_6 \left\| f \int_\Omega \right.
&\quad \left. \left\| \widetilde{S}_{a,-}^{-1,0} W_2(\Omega, \Gamma_l) \right\| \left\| |u| \overset{\circ}{\rightarrow} W_2^{\widetilde{A}} \right. \right.
&\quad \left. \left. (\Omega; \Gamma_r, T) \right\| \right\|.
\end{split}
\end{equation}
```

One further example involving `split` and `align`. To obtain unnumbered equations use the `align*` environment instead.

$$\begin{aligned}
|I_1| &= \left| \int_{\Omega} gRu \, d\Omega \right| \\
&\leq C_3 \left[ \int_{\Omega} \left( \int_a^x g(\xi, t) \, d\xi \right)^2 d\Omega \right]^{1/2} \\
&\quad \times \left[ \int_{\Omega} \left\{ u_x^2 + \frac{1}{k} \left( \int_a^x cu_t \, d\xi \right)^2 \right\} c\Omega \right]^{1/2} \\
&\leq C_4 \left\| f \left| \tilde{S}_{a,-}^{-1,0} W_2(\Omega, \Gamma_l) \right| \right\| \left\| |u| \overset{\circ}{\rightarrow} W_2^{\tilde{A}}(\Omega; \Gamma_r, T) \right\|.
\end{aligned} \tag{8.49}$$

$$\begin{aligned}
|I_2| &= \left| \int_0^T \psi(t) \left\{ u(a, t) - \int_{\gamma(t)}^a \frac{d\theta}{k(\theta, t)} \int_a^{\theta} c(\xi) u_t(\xi, t) \, d\xi \right\} dt \right| \\
&\leq C_6 \left\| f \int_{\Omega} \left| \tilde{S}_{a,-}^{-1,0} W_2(\Omega, \Gamma_l) \right| \right\| \left\| |u| \overset{\circ}{\rightarrow} W_2^{\tilde{A}}(\Omega; \Gamma_r, T) \right\|.
\end{aligned} \tag{8.50}$$

The input for the above formulae is:

```

\begin{align}
\begin{split}
|I_1| &= \left| \int_{\Omega} gRu \, d\Omega \right| && \backslash\backslash \\
&\leq C_3 \left[ \int_{\Omega} \left( \int_a^x g(\xi, t) \, d\xi \right)^2 d\Omega \right]^{1/2} && \backslash\backslash \\
&\quad \times \left[ \int_{\Omega} \left\{ u_x^2 + \frac{1}{k} \left( \int_a^x cu_t \, d\xi \right)^2 \right\} c\Omega \right]^{1/2} && \backslash\backslash \\
&\leq C_4 \left\| f \left| \tilde{S}_{a,-}^{-1,0} W_2(\Omega, \Gamma_l) \right| \right\| \left\| |u| \overset{\circ}{\rightarrow} W_2^{\tilde{A}}(\Omega; \Gamma_r, T) \right\|. && \backslash\backslash
\end{split} \\
\end{split} \tag{eq:A} \\
\begin{split}
|I_2| &= \left| \int_0^T \psi(t) \left\{ u(a, t) \right. \right. \\
&\quad \left. \left. - \int_{\gamma(t)}^a \frac{d\theta}{k(\theta, t)} \int_a^{\theta} c(\xi) u_t(\xi, t) \, d\xi \right\} dt \right| && \backslash\backslash \\
&\leq C_6 \left\| f \int_{\Omega} \left| \tilde{S}_{a,-}^{-1,0} W_2(\Omega, \Gamma_l) \right| \right\| \left\| |u| \overset{\circ}{\rightarrow} W_2^{\tilde{A}}(\Omega; \Gamma_r, T) \right\|. && \backslash\backslash
\end{split} \\
\end{split} \\
\end{align}

```

### 8.7.2 The multiline Environment

Numbered version:

$$\left| \begin{aligned} \int_a^b \left\{ \int_a^b [f(x)^2 g(y)^2 + f(y)^2 g(x)^2] - 2f(x)g(x)f(y)g(y) dx \right\} dy \\ = \int_a^b \left\{ g(y)^2 \int_a^b f^2 + f(y)^2 \int_a^b g^2 - 2f(y)g(y) \int_a^b fg \right\} dy \quad (8.51) \end{aligned} \right|$$

This was obtained with the lines shown below.

```
\begin{multiline}\label{eq:E}
  \int_a^b \biggl\{ \int_a^b [ f(x)^2 g(y)^2 + f(y)^2 g(x)^2 ]
    -2f(x) g(x) f(y) g(y) \,dx \biggr\} \,dy \quad \\\
  =\int_a^b \biggl\{ g(y)^2 \int_a^b f^2 + f(y)^2
    \int_a^b g^2 - 2f(y)g(y) \int_a^b fg \biggr\} \,dy
\end{multiline}
```

An unnumbered version of the above is obtained with the same input, except the `multiline` environment is replaced by `multiline*`.

$$\left| \begin{aligned} \int_a^b \left\{ \int_a^b [f(x)^2 g(y)^2 + f(y)^2 g(x)^2] - 2f(x)g(x)f(y)g(y) dx \right\} dy \\ = \int_a^b \left\{ g(y)^2 \int_a^b f^2 + f(y)^2 \int_a^b g^2 - 2f(y)g(y) \int_a^b fg \right\} dy \end{aligned} \right|$$

And now an unnumbered version numbered with a `\tag*` command.

$$\left| \begin{aligned} \int_a^b \left\{ \int_a^b [f(x)^2 g(y)^2 + f(y)^2 g(x)^2] - 2f(x)g(x)f(y)g(y) dx \right\} dy \\ = \int_a^b \left\{ g(y)^2 \int_a^b f^2 + f(y)^2 \int_a^b g^2 - 2f(y)g(y) \int_a^b fg \right\} dy \quad [a] \end{aligned} \right|$$

This was generated with:

```
\begin{multiline*}\tag*{[a]} \dots \end{multiline*}
```

This is the same display, but with `\multlinegap` set to zero. Notice that the space on the left of the first line does not change, because of the equation number, while the second line is pushed over to the right margin.

$$\left| \begin{aligned} \int_a^b \left\{ \int_a^b [f(x)^2 g(y)^2 + f(y)^2 g(x)^2] - 2f(x)g(x)f(y)g(y) dx \right\} dy \\ = \int_a^b \left\{ g(y)^2 \int_a^b f^2 + f(y)^2 \int_a^b g^2 - 2f(y)g(y) \int_a^b fg \right\} dy \quad [a] \end{aligned} \right|$$

This was generated with:

```
{\setlength{\multlinegap}{0pt}
\begin{multline*}\tag*[a] ... \end{multline*}}
```

### 8.7.3 The gather Environment

Numbered version with `\notag` on the second line:

$$D(a, r) \equiv \{z \in \mathbf{C}: |z - a| < r\}, \quad (8.52)$$

$$\operatorname{seg}(a, r) \equiv \{z \in \mathbf{C}: \Im z = \Im a, |z - a| < r\},$$

$$c(e, \theta, r) \equiv \{(x, y) \in \mathbf{C}: |x - e| < y \tan \theta, 0 < y < r\}, \quad (8.53)$$

$$C(E, \theta, r) \equiv \bigcup_{e \in E} c(e, \theta, r). \quad (8.54)$$

This was generated with:

```
\begin{gather}
D(a,r) \equiv \{ z \in \mathbf{C}: |z-a| < r \}, \quad \\\
\operatorname{seg}(a,r) \equiv \{ z \in \mathbf{C}: \\
\quad \Im z = \Im a, \ |z-a| < r \}, \quad \notag \\\
c(e,\theta,r) \equiv \{ (x,y) \in \mathbf{C}: \\
\quad |x-e| < y \tan \theta, \ 0 < y < r \}, \quad \\\
C(E,\theta,r) \equiv \bigcup_{e \in E} c(e,\theta,r). \\
\end{gather}
```

### 8.7.4 The align Environment

Numbered version:

$$\gamma_x(t) = (\cos tu + \sin tx, v), \quad (8.55)$$

$$\gamma_y(t) = (u, \cos tv + \sin ty), \quad (8.56)$$

$$\gamma_z(t) = \left( \cos tu + \frac{\alpha}{\beta} \sin tv, -\frac{\beta}{\alpha} \sin tu + \cos tv \right). \quad (8.57)$$

This was produced using the following input:

```
\begin{align}
\gamma_x(t) &= (\cos tu + \sin tx, v), \quad \\\
\gamma_y(t) &= (u, \cos tv + \sin ty), \quad \\\
\gamma_z(t) &= \left( \cos tu + \frac{\alpha}{\beta} \sin tv, \\
&\quad - \frac{\beta}{\alpha} \sin tu + \cos tv \right). \\
\end{align}
```

Unnumbered version:

$$\left. \begin{aligned} \gamma_x(t) &= (\cos tu + \sin tv, v), \\ \gamma_y(t) &= (u, \cos tv + \sin ty), \\ \gamma_z(t) &= \left( \cos tu + \frac{\alpha}{\beta} \sin tv, -\frac{\beta}{\alpha} \sin tu + \cos tv \right). \end{aligned} \right\}$$

This was generated using the following construct:

```
\begin{align*} ... \end{align*}
```

### 8.7.5 Using the align and split Environments within gather

When using the `align` environment within the `gather` environment, one or the other, or both, should be unnumbered (using the `*` form), since having numbering for both the outer and inner environment would not be meaningful.

Automatically numbered `gather` with `split` and `align*`:

$$\left. \begin{aligned} \varphi(x, z) &= z - \gamma_{10}x - \sum_{m+n \geq 2} \gamma_{mn}x^m z^n \\ &= z - Mr^{-1}x - \sum_{m+n \geq 2} Mr^{-(m+n)}x^m z^n \end{aligned} \right\} (8.58)$$

$$\begin{aligned} \zeta^0 &= (\xi^0)^2, \\ \zeta^1 &= \xi^0 \xi^1 \end{aligned}$$

Here the `split` environment gets a number from the outer `gather` environment; numbers for individual lines of the `align*` are suppressed because of the star.

```
\begin{gather}
\begin{split}
\varphi(x, z)
&= z - \gamma_{10} x - \sum_{m+n \geq 2} \gamma_{mn} x^m z^n \\
&= z - M r^{-1} x - \sum_{m+n \geq 2} M r^{-(m+n)} x^m z^n
\end{split}
\end{gather} \quad \ll[6pt]
```

```
\begin{align*}
\zeta^0 &= (\xi^0)^2, \quad \ll \\
\zeta^1 &= \xi^0 \xi^1
\end{align*}
\end{gather}
```

Shown below, is the `*`-ed form of `gather` with the non-`*`-ed form of `align`.



$x = y$	by (8.49)	(8.63)
$x' = y'$	by (8.61)	(8.64)
$x + x' = y + y'$	by Axiom 1.	(8.65)

This example was obtained with the commands below:

```
\begin{alignat}{2}
x      &= y      && \quad \text{by (\ref{eq:A})}\label{eq:C} \\
x'     &= y'     && \quad \text{by (\ref{eq:B})}\label{eq:D} \\
x + x' &= y + y' && \quad \text{by Axiom 1.} \\
\end{alignat}
```

The expanded version, `flalign`:

$x = y$	by (8.63)	(8.66)
$x' = y'$	by (8.64)	(8.67)
$x + x' = y + y'$	by Axiom 1.	(8.68)

This was generated using the following construct:

```
\begin{flalign} ... \end{flalign}
```

## 8.8 Extensions to the theorem Environment

( $\mathcal{L}$  58, 174)

$\mathcal{A}\mathcal{M}\mathcal{S}$ - $\mathcal{L}\mathcal{A}\mathcal{T}\mathcal{E}\mathcal{X}$  comes with the `amsthm` package, which extends  $\mathcal{L}\mathcal{A}\mathcal{T}\mathcal{E}\mathcal{X}$ 's `\newtheorem` command. Rather than describe `amsthm` (see, for instance, the section “Proclamations” in Grätzer’s book [?] for more details) we will give some details about the `theorem` package, developed by Frank Mittelbach [?]. It also offers an extension of the  $\mathcal{L}\mathcal{A}\mathcal{T}\mathcal{E}\mathcal{X}$  theorem mechanism by allowing the layout of theorems to be manipulated by specifying a style.

In the present context the word “theorem” is used for any kind of labeled enunciations, often set off from the main text by extra space and a font change. Theorems, corollaries, conjectures, definitions, and remarks are all instances of “theorems.” The header of these structures is composed of a label (such as `THEOREM` or `REMARK`) and a number, which serializes an item in the sequence of items with the same label.

Often it is necessary, in order to satisfy the requirements of different mathematics journals, to customize the layout of the theorem environment. Additionally, different formats may be needed to differentiate the “sort of theorem”: e.g., remarks and definitions are set in roman, while italic is employed for main theorems.

### 8.8.1 Defining New Theorem Environments

As in the original L<sup>A</sup>T<sub>E</sub>X version, the command `\newtheorem` defines a new “theorem-like structure.” Two required arguments name the new environment and give the text to be typeset with each instance of the new environment, while an optional argument determines how the environment is enumerated:

```
\newtheorem{env-name}{label-text}
```

The above `\newtheorem` command defines the *env-name* environment and its printed name will be *label-text*. It uses its own counter.

```
\newtheorem{env2-name}[env-name]{label-text2}
```

The above `\newtheorem` command defines the *env2-name* environment, and its printed name will be *label-text2*. It uses the same counter as theorem set *env-name*.

```
\newtheorem{env3-name}{label-text3}[section]
```

The above variant defines the *env3-name* environment and its printed name is *label-text3*. Its counter is enumerated within the counter *section*, that is, with every new `\section` the enumeration starts again with one, and the enumeration is composed from the section number and the theorem counter itself.

```
\theoremstyle{style}
```

The `\theoremstyle` command can define the layout of various, or all, theorem sets. It should be noted that any theorem set defined by `\newtheorem` is typeset in the `\theoremstyle` that is current at the time of the definition.

Thus, the following

```
\theoremstyle{break}      \newtheorem{Cor}{Corollary}
\theoremstyle{plain}      \newtheorem{Exa}{Example}[section]
```

leads to the result that the set `Cor` is formatted in the style `break`, while the set `Exa` and all the following ones are formatted in the style `plain`, unless another `\theoremstyle` follows. Since the definitions installed by `\newtheorem` are global, you can also limit `\theoremstyle` locally by grouping braces.

```
\theorembodyfont{font-declarations}
```

The choice of the font for the theorem body is completely independent of the chosen `\theoremstyle`; this has proven to be very advantageous. For example,

```
{\theorembodyfont{\rmfamily} \newtheorem{Rem}{Remark}}
```

<code>plain</code>	Emulates the original L <sup>A</sup> T <sub>E</sub> X definition, except that additionally the parameters <code>\theorempreskipamount</code> and <code>\theorempostskipamount</code> are used.
<code>break</code>	In this style, the theorem header is followed by a line break.
<code>marginbreak</code>	The theorem number is set in the margin, and there is a line break as in <code>break</code> .
<code>changebreak</code>	Like <code>break</code> , but with header number and text interchanged.
<code>change</code>	Header number and text are interchanged, without a line break.
<code>margin</code>	The number is set in the left margin, without a line break.

Table 8.22: List of existing theorem styles

All styles (except `plain`) select `\normalfont\slshape` as the default for `\theorembodyfont`.

defines a theorem set `Rem`, which will be set in `\rmfamily` in the current layout (which in our example is `plain`). As with `\theoremstyle`, the `\theorembodyfont` chosen is that which is current at the time of `\newtheorem`. If `\theorembodyfont` is not specified or you define `\theorembodyfont{}`, then the font used will be defined by `\theoremstyle`.

`\theoremheaderfont{font-declarations}`

It is also possible to customize the font used for the theorem headers. This is, however, a global declaration and, therefore, there should be at most one `\theoremheaderfont` command in the preamble. If it is actually necessary to have different header fonts, you will have to define new theorem styles (substituting the desired font).

Two additional parameters affect the vertical space around the theorem environments: `\theorempreskipamount` and `\theorempostskipamount` define, respectively, the spacing before and after such an environment. These parameters apply to all theorem sets and can be manipulated with the ordinary length macros. They are rubber lengths, and therefore can contain `plus` and `minus` parts. These parameters are set using the `\setlength` command.

The commands to define theorem sets, as described in this section, can only be placed in the document preamble or in a package file.

Theorem styles, which exist to date, are shown in table 8.22

## 8.8.2 Examples of the Definition and Use of Theorems

Suppose that the preamble contains the declarations:

```

\theoremstyle{break}      \newtheorem{Cor}{Corollary}
\theoremstyle{plain}     \newtheorem{Exa}{Example}[section]

```

```

{\theorembodyfont{\rmfamily} \newtheorem{Rem}{Remark}}
\theoremstyle{marginbreak} \newtheorem{Lem}[Cor]{Lemma}
\theoremstyle{change}
\theorembodyfont{\itshape} \newtheorem{Def}[Cor]{Definition}

\theoremheaderfont{\scshape}

```

Then the typical examples below show the typeset output resulting from their use.

<p>COROLLARY 1 <i>This is a sentence typeset in the theorem environment Cor.</i></p>	<pre> \begin{Cor}   This is a sentence typeset in the theorem   environment \Lenv{Cor}. \end{Cor} </pre>
<p>EXAMPLE 8.8.1 <i>This is a sentence typeset in the theorem environment Exa.</i></p>	<pre> \begin{Exa}   This is a sentence typeset in the theorem   environment \Lenv{Exa}. \end{Exa} </pre>
<p>REMARK 1 This is a sentence typeset in the theorem environment Rem.</p>	<pre> \begin{Rem}   This is a sentence typeset in the theorem   environment \Lenv{Rem}. \end{Rem} </pre>
<p>2 LEMMA (BEN USER) <i>This is a sentence typeset in the theorem environment Lem.</i></p>	<pre> \begin{Lem}[Ben User]   This is a sentence typeset in the theorem   environment \Lenv{Lem}. \end{Lem} </pre>
<p>3 DEFINITION (VERY IMPRESSIVE DEFINITION) <i>This is a sentence typeset in the theorem environment Def.</i></p>	<pre> \begin{Def}[Very impressive Definition]   This is a sentence typeset in the theorem   environment \Lenv{Def}. \end{Def} </pre>

The last two examples show the effect of the optional argument to a theorem environment (it is typeset in parentheses right after the label).

### 8.8.3 Special Considerations

The theorem header and body are implemented as a single unit. This means that the `\theoremheaderfont` will inherit characteristics of the `\theorembodyfont` if the NFSS is being used. Thus, if, for example, `\theorembodyfont` is `\itshape`

and `\theoremheaderfont` is `\bfseries` the font selected for the header will have the characteristics “bold extended italic.” If this is not desired you should set it to something like `\theoremheaderfont{\normalfont\bfseries}`. That is, you should supply all the necessary font information explicitly. See chapter 7 for more details about how to do that.

## 8.9 Mathematical Style Parameters

This section explains how you can globally control the style of your mathematical formulae, and how you can modify the size of certain (sub)formula elements.

### 8.9.1 Controlling the Size of Characters

Letters and mathematical symbols sometimes get smaller when they appear in fractions, superscripts, or subscripts. In fact,  $\TeX$  has eight different styles in which it can treat formulae, namely:

$D, D'$	<code>\displaystyle</code>	formulae displayed on lines by themselves
$T, T'$	<code>\textstyle</code>	formulae embedded in the text
$S, S'$	<code>\scriptstyle</code>	formulae used as super- or subscripts
$SS, SS'$	<code>\scriptscriptstyle</code>	second- and higher-order super- or subscripts

The accented symbols represent the so-called *cramped* styles, which are similar to the normal styles except that exponents are not raised so much.  $\TeX$  also uses three different type sizes for mathematics, namely: text size, script size, and scriptscript size.

A formula set inside text (between a  $\$$  pair, or between `\(. . .\)`) is typeset using text style (style  $T$ ). A formula on a line by itself, e.g., entered between `\[. . .\]`, will be typeset in display style (style  $D$ ). The size of the different parts of a formula can be determined according to the following scheme:

A symbol in style	will be typeset in	(example)
$D, D', T, T'$	text size	(text size)
$S, S'$	script size	(script size)
$SS, SS'$	scriptscript size	(scriptscript size)

The kind of style used in mathematics formulae is as follows:

style	superscript	subscript	numerator	denominator
$D$	$S$	$S'$	$T$	$T'$
$D'$	$S'$	$S'$	$T'$	$T'$
$T$	$S$	$S'$	$S$	$S'$
$T'$	$S'$	$S'$	$S'$	$S'$
$S, SS$	$SS$	$SS'$	$SS$	$SS'$
$S', SS'$	$SS'$	$SS'$	$SS'$	$SS'$

The last two columns describe the style used in the numerator or denominator of a fraction. An example of the various styles can be seen in the continued fraction below (see also section 8.3.16):

$$b^0 + \frac{a^1}{b_1 + \frac{a^2}{b_2 + \frac{a^3}{b_3}}}$$

```

\normalsize
\[ b^0 + \frac{a^1}{b_1 + \frac{a^2}{b_2 + \frac{a^3}{b_3}}}
\]

```

In the formula above the  $b$  of  $b^0$  is in style  $D$ , with the 0 in style  $S$ ; the  $a$  and  $b$  of  $a^1$  and  $b_1$  are in style  $T$  and  $T'$ , respectively, with the exponent 1 in style  $S$  and the subscript 1 in style  $S'$ ; the  $a$  and  $b$  of  $a^2$  and  $b_2$  are both in style  $S'$ , with the exponent and subscript in style  $SS'$ ; finally everything in  $a^3$  and  $b_3$  is in style  $SS'$ .

You can give a nicer look to the above example by deciding which style is to be used in each case. Note that to save typing, we define the abbreviation  $\backslash D$  for the  $\backslash displaystyle$  command.

$$b^0 + \frac{a^1}{b_1 + \frac{a^2}{b_2 + \frac{a^3}{b_3}}}$$

```

\newcommand{\D}{\displaystyle}
\normalsize
\[ b^0 + \frac{a^1}{\D b_1 + \frac{a^2}{\D b_2 + \frac{a^3}{b_3}}}
\]

```

### 8.9.2 L<sup>A</sup>T<sub>E</sub>X Math Style Parameters

Because L<sup>A</sup>T<sub>E</sub>X uses much of the mathematical machinery from T<sub>E</sub>X, we briefly (L 170)

describe the mathematical style parameters that L<sup>A</sup>T<sub>E</sub>X uses to typeset formulae.

All these are length parameters which you can redefine with the  $\backslash setlength$  or

$\backslash addtolength$  commands (see section A.1.4 on page 474). Moreover, two stan-

dard options,  $leqno$  and  $fleqn$ , control the numbering and alignment of formulae. (L 82)

The option `fleqn` causes formulae to be aligned on the left, a fixed distance from the left margin (see `\mathindent` below), instead of being centered.

The option `leqno` causes formula numbers to appear on the left instead of at the right (see section 8.6.6 on page 252).

In the list of mathematics style parameters below, all lengths (except `\jot` and `\arraycolsep`) are rubber lengths. With the option `fleqn`, the four `display` lengths are made equal to the list defining length `\topsep`, to which the value of `\partopsep` is added if the display starts a paragraph (see figure 3.5 on page 64). The four parameters `\abovedisplay...` and `\belowdisplay...` below depend on the current font size. For this reason they cannot be modified in the preamble of the document using `\setlength`, but they must be changed by modifying `\normalsize`, etc.

`\arraycolsep` This gives half the width of the horizontal space between columns in an `array` environment (default value `5pt`, see also section 5.3.2).

`\jot` This is the extra vertical space that is added between rows in an `eqnarray` or `eqnarray*` environment (default value `3pt`).

`\mathindent` This defines the indentation from the left margin of displayed formulae for the `fleqn` option (the default value is equal to the indentation of a first level list, i.e., `2.5em`, and is defined by the option `fleqn`).

`\abovedisplayskip` This specifies the extra space left above a long displayed formula, except with the option `fleqn`, where `\topsep` is used. A long formula is one that lies closer to the left margin than does the end of the preceding line (default value `12pt plus 3pt minus 9pt`).

`\belowdisplayskip` This specifies the extra space left below a long displayed formula, except with the option `fleqn`, where `\topsep` is used (default value `12pt plus 3pt minus 9pt`).

`\abovedisplayshortskip` This specifies the extra space left above a short displayed formula, except with the option `fleqn`, where `\topsep` is used. A short formula is one which starts to the right of where the preceding line ends (default value `0pt plus 3pt`).

`\belowdisplayshortskip` This specifies the extra space left below a short displayed formula, except with the option `fleqn`, where `\topsep` is used (default value `7pt plus 3pt minus 4pt`).