

Symbols & Logical Syntax in L<sup>A</sup>T<sub>E</sub>X

Lewis Britton

Greek & Hebrew Characters

Alphabetical Letters					
A, α	<code>\Alpha, \alpha</code>	I, ι	<code>\Iota, \iota</code>	P, ρ, ϱ	<code>\Rho, \rho, \varrho</code>
B, β	<code>\Beta, \beta</code>	K, κ, ⋈	<code>\Kappa, \kappa, \varkappa</code>	Σ, σ, ϸ	<code>\Sigma, \sigma, \varsigma</code>
Γ, γ	<code>\Gamma, \gamma</code>	Λ, λ	<code>\Lambda, \lambda</code>	T, τ	<code>\Tau, \tau</code>
Δ, δ	<code>\Delta, \delta</code>	M, μ	<code>\Mu, \mu</code>	Υ, υ	<code>\Upsilon, \upsilon</code>
E, ε, ε	<code>\Epsilon, \epsilon, \varepsilon</code>	N, ν	<code>\Nu, \nu</code>	Φ, φ, ϕ	<code>\Phi, \phi, \varphi</code>
Z, ζ	<code>\Zeta, \zeta</code>	Ξ, ξ	<code>\Xi, \xi</code>	X, χ, ϳ	<code>\Chi, \chi</code>
H, η	<code>\Eta, \eta</code>	O, o	<code>\Omicron, \omicron</code>	Ψ, ψ	<code>\Psi, \psi</code>
Θ, θ, ϑ	<code>\Theta, \theta, \vartheta</code>	Π, π, ϖ	<code>\Pi, \pi, \varpi</code>	Ω, ω	<code>\Omega, \omega</code>

Miscellaneous Characters & Punctuation					
ƒ	<code>\digamma</code>	℄	<code>\complement</code>	∠	<code>\angle</code>
ℵ	<code>\aleph</code>	ℓ	<code>\ell</code>	∠	<code>\measuredangle</code>
Ɑ	<code>\beth</code>	⋈	<code>\eth</code>	∠	<code>\sphericalangle</code>
Ɱ	<code>\daleth</code>	ℎ	<code>\hbar</code>	√	<code>\surd</code>
Ɐ	<code>\gimel</code>	ℎ	<code>\hslash</code>	ℵ	<code>\natural</code>
ι	<code>\imath</code>	⊤	<code>\top</code>	♯	<code>\sharp</code>
ⱱ	<code>\jmath</code>	⊥	<code>\bot</code>	♭	<code>\flat</code>
∇	<code>\nabla</code>	§	<code>\S</code>	Δ	<code>\vartriangle</code>
△	<code>\triangle</code>	∅	<code>\varnothing</code>	▽	<code>\triangledown</code>
▲	<code>\blacktriangle</code>	■	<code>\blacksquare</code>	↘	<code>\diagdown</code>
▼	<code>\blacktriangledown</code>	◆	<code>\blacklozenge</code>	↗	<code>\diagup</code>
				℔	<code>\Im</code>
				ℜ	<code>\Re</code>
				ℳ	<code>\mho</code>
				℘	<code>\wp</code>
				℔	<code>\Bbbk</code>
				∅	<code>\emptyset</code>
				∞	<code>\infty</code>
				□, ◻	<code>\Box, \square</code>
				◇	<code>\Diamond</code>
				∃	<code>\exists</code>
				∄	<code>\nexists</code>
				⊃	<code>\Game</code>
				⊄	<code>\Finv</code>
				∂	<code>\partial</code>
				™, ©	<code>\trademark, \copyright</code>
				£, \$	<code>\pounds, \\$</code>
				◇, ◇	<code>\diamondsuit, \lozenge</code>
				♥	<code>\heartsuit</code>
				♣	<code>\clubsuit</code>
				♠	<code>\spadesuit</code>
				★	<code>\bigstar</code>

Text Mode Miscellaneous Characters & Punctuation

ó	<code>\' {o}</code>	o	<code>\b {o}</code>	ö	<code>\v o</code>	Ø, ø	<code>\O, \o</code>	¶	<code>\P</code>	£, \$	<code>\pounds, \\$</code>
ò	<code>\' {o}</code>	ò	<code>\. {o}</code>	ơ	<code>\d o</code>	Å, å	<code>\AA, \aa</code>	§	<code>\S</code>	!, ?	<code>!, ?</code>
ô	<code>\" {o}</code>	ơ	<code>\d {o}</code>	õ	<code>\r o</code>	Æ, æ	<code>\AE, \ae</code>	†	<code>\dag</code>	., ,	<code>., ,</code>
ô	<code>\^ {o}</code>	q	<code>\c {o}</code>	ö	<code>\H o</code>	ß	<code>\ss</code>	‡	<code>\ddag</code>	‘, ’	<code>‘, ’</code>
ô	<code>\~ {o}</code>	ö	<code>\u {o}</code>	ơ	<code>\t o</code>	ı	<code>\i</code>	™, ©	<code>\trademark, \copyright</code>	“, ”	<code>“, ” or "</code>
ô	<code>\= {o}</code>	ö	<code>\H {o}</code>	oo	<code>\t {oo}</code>	j	<code>\j</code>	®, ®	<code>\textregistered, \circledR</code>	∴, ∵	<code>∴, ∵</code>

Basic Math Mode

Alphabets					
<i>XYX xyz</i>	<code>XYX\ xyz</code>	<b>XYZ xyz</b>	<code>\mathbf{XYZ\ xyz}</code>	<i>XYZ</i>	<code>\mathbb{XYZ}</code>
<i>XYZ xyz</i>	<code>\mathnormal{XYZ\ xyz}</code>	XYZ xyz	<code>\mathsf{XYZ\ xyz}</code>	<i>XYZ</i>	<code>\mathcal{XYZ}</code>
<i>XYZ xyz</i>	<code>\mathit{XYZ\ xyz}</code>	XYZ xyz	<code>\mathtt{XYZ\ xyz}</code>	<i>XYZ</i>	<code>\mathfrak{XYZ}</code>
<i>XYZ xyz</i>	<code>\mathrm{XYZ\ xyz}</code>				

Spacing					
<i>xyz</i>	<code>xyz</code>	Default math	<i>dad</i>	<code>a\!b\mspace{-3mu}c\negthinspace d</code>	Neg. 3mu ‘thin’
<i>x y z</i>	<code>x\ y\ z</code>	Expanded	<i>dad</i>	<code>a\negmedspace b\mspace{-4mu}c\negmedspace d</code>	Neg. 4mu ‘medium’
<i>sin x cos y</i>	<code>\sin x\cos y</code>	Operator	<i>dad</i>	<code>a\negthickspace b\mspace{-5mu}c\negthickspace d</code>	Neg. 5mu ‘thick’
<i>a b c d</i>	<code>a\,b\mspace{3mu}c\thinspace d</code>	3mu ‘thin’	<i>a b</i>	<code>a\phantom{xxx}b</code>	Width of ‘xxx’
<i>a b c d</i>	<code>a\:b\mspace{4mu}c\medspace d</code>	4mu ‘medium’			
<i>a b c d</i>	<code>a\:b\mspace{5mu}c\thickspace d</code>	5mu ‘thick’			

Math Accents & Constructs

Note that most basic accents can be stacked. For example, `\acute{\acute{x}}` yields  $\acute{\acute{x}}$ . Or, `\acute{\tilde{x}}` yields  $\acute{\tilde{x}}$ .

$\acute{x}$	<code>\acute{x}</code>	$\dot{x}$	<code>\dot{x}</code>	$\overline{xyz}$	<code>\overline{xyz}</code>	$\xleftarrow[abc]{xyz}$	<code>\xleftarrow[abc]{xyz}</code>	$\sum^K$	<code>\overset{K}{\sum}</code>
$\grave{x}$	<code>\grave{x}</code>	$\ddot{x}$	<code>\ddot{x}</code>	$\underline{xyz}$	<code>\underline{xyz}</code>	$\xrightarrow[abc]{xyz}$	<code>\xrightarrow[abc]{xyz}</code>	$\sum_{k=1}$	<code>\underset{k=1}{\sum}</code>
$\bar{x}$	<code>\bar{x}</code>	$\check{x}$	<code>\check{x}</code>	$\overrightarrow{xyz}$	<code>\overrightarrow{xyz}</code>	$\overbrace{xyz}$	<code>\overbrace{xyz}</code>	$\sqrt{x}$	<code>\sqrt{x}</code>
$\hat{x}$	<code>\hat{x}</code>	$\vec{x}$	<code>\vec{x}</code>	$\overleftarrow{xyz}$	<code>\overleftarrow{xyz}</code>	$\underbrace{xyz}$	<code>\underbrace{xyz}</code>	$\sqrt[n]{x}$	<code>\sqrt[n]{x}</code>
$\tilde{x}$	<code>\tilde{x}</code>	$\widehat{xyz}$	<code>\widehat{xyz}</code>	$\overleftrightharrow{xyz}$	<code>\overleftrightharrow{xyz}</code>	$f, f', f'$	<code>f, f', f\prime</code>		
$\breve{x}$	<code>\breve{x}</code>	$\widetilde{xyz}$	<code>\widetilde{xyz}</code>	$\frac{abc}{xyz}$	<code>\frac{abc}{xyz}</code>	$\sum_{y\sum_k^j}$	<code>\sideset{y^x}{_k^j}\sum</code>		

Binary Relations

Note that you can produce according negations by either adding the `\not` command as a prefix or ordinarily by preceding the commands with ‘n’. For example, `\not=` or `\neq` turns  $=$  to  $\neq$ . This rule also holds for arrows, which will be seen later.

Standard Relations

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

Additional Relations

$\lessdot$	<code>\lessdot</code>	$\gtrdot$	<code>\gtrdot</code>	$\subseteq$	<code>\subseteq</code>	$\supseteq$	<code>\supseteq</code>	$\thicksim$	<code>\thicksim</code>
$\leqslant$	<code>\leqslant</code>	$\geqslant$	<code>\geqslant</code>	$\Subset$	<code>\Subset</code>	$\Supset$	<code>\Supset</code>	$\thickapprox$	<code>\thickapprox</code>
$\eqslantless$	<code>\eqslantless</code>	$\eqslantgtr$	<code>\eqslantgtr</code>	$\sqsubset$	<code>\sqsubset</code>	$\sqsupset$	<code>\sqsupset</code>	$\approxeq$	<code>\approxeq</code>
$\leqq$	<code>\leqq</code>	$\geqq$	<code>\geqq</code>	$\therefore$	<code>\therefore</code>	$\because$	<code>\because</code>	$\backsimeq$	<code>\backsimeq</code>
$\lll$ or $\llless$	<code>\lll</code> or <code>\llless</code>	$\gggtr$	<code>\gggtr</code>	$\cdot$	<code>\cdot</code>	$\shortparallel$	<code>\shortparallel</code>	$\backsim$	<code>\backsim</code>
$\lessapprox$	<code>\lessapprox</code>	$\gtrapprox$	<code>\gtrapprox</code>	$\smallsmile$	<code>\smallsmile</code>	$\smallfrown$	<code>\smallfrown</code>	$\vdash$	<code>\vdash</code>
$\lessgtr$	<code>\lessgtr</code>	$\gtrless$	<code>\gtrless</code>	$\triangleleft$	<code>\triangleleft</code>	$\triangleright$	<code>\triangleright</code>	$\vdash$	<code>\vdash</code>
$\lesseqgtr$	<code>\lesseqgtr</code>	$\gtreqless$	<code>\gtreqless</code>	$\trianglelefteq$	<code>\trianglelefteq</code>	$\trianglerighteq$	<code>\trianglerighteq</code>	$\Vdash$	<code>\Vdash</code>
$\lesseqqgtr$	<code>\lesseqqgtr</code>	$\gtreqqless$	<code>\gtreqqless</code>	$\blacktriangleleft$	<code>\blacktriangleleft</code>	$\blacktriangleright$	<code>\blacktriangleright</code>	$\backepsilon$	<code>\backepsilon</code>
$\precurlyeq$	<code>\precurlyeq</code>	$\succcurlyeq$	<code>\succcurlyeq</code>	$\doteqdot$ or $\Doteq$	<code>\doteqdot</code> or <code>\Doteq</code>	$\eqcirc$	<code>\eqcirc</code>	$\varpropto$	<code>\varpropto</code>
$\curlyeqprec$	<code>\curlyeqprec</code>	$\curlyeqsucc$	<code>\curlyeqsucc</code>	$\risingdotseq$	<code>\risingdotseq</code>	$\fallingdotseq$	<code>\fallingdotseq</code>	$\between$	<code>\between</code>
$\precsim$	<code>\precsim</code>	$\succsim$	<code>\succsim</code>	$\circeq$	<code>\circeq</code>	$\tiraqleeq$	<code>\tiraqleeq</code>	$\pitchfork$	<code>\pitchfork</code>
$\precapprox$	<code>\precapprox</code>	$\succapprox$	<code>\succapprox</code>	$\bumpeq$	<code>\bumpeq</code>	$\Bumpeq$	<code>\Bumpeq</code>		

Negated Relations

$\nless$	<code>\nless</code>	$\ngtr$	<code>\ngtr</code>	$\subsetneq$	<code>\subsetneq</code>	$\supsetneq$	<code>\supsetneq</code>
$\lneq$	<code>\lneq</code>	$\gneq$	<code>\gneq</code>	$\varsubsetneq$	<code>\varsubsetneq</code>	$\varsupsetneq$	<code>\varsupsetneq</code>
$\nleq$	<code>\nleq</code>	$\ngeq$	<code>\ngeq</code>	$\nsubseteq$	<code>\nsubseteq</code>	$\nsupseteq$	<code>\nsupseteq</code>
$\nleqslant$	<code>\nleqslant</code>	$\ngeqslant$	<code>\ngeqslant</code>	$\nsubsetneqq$	<code>\nsubsetneqq</code>	$\nsupsetneqq$	<code>\nsupsetneqq</code>
$\lneqq$	<code>\lneqq</code>	$\gneqq$	<code>\gneqq</code>	$\varsubsetneqq$	<code>\varsubsetneqq</code>	$\varsupsetneqq$	<code>\varsupsetneqq</code>
$\lvertneqq$	<code>\lvertneqq</code>	$\gvertneqq$	<code>\gvertneqq</code>	$\nsubseteqq$	<code>\nsubseteqq</code>	$\nsupseteqq$	<code>\nsupseteqq</code>
$\nleqq$	<code>\nleqq</code>	$\ngeqq$	<code>\ngeqq</code>	$\nmid$	<code>\nmid</code>	$\nparallel$	<code>\nparallel</code>
$\lnsim$	<code>\lnsim</code>	$\gnsim$	<code>\gnsim</code>	$\nshortmid$	<code>\nshortmid</code>	$\nshortparallel$	<code>\nshortparallel</code>
$\lnapprox$	<code>\lnapprox</code>	$\gnapprox$	<code>\gnapprox</code>	$\nsim$	<code>\nsim</code>	$\ncong$	<code>\ncong</code>
$\nprec$	<code>\nprec</code>	$\nsucc$	<code>\nsucc</code>	$\nvdash$	<code>\nvdash</code>	$\nvDash$	<code>\nvDash</code>
$\npreceq$	<code>\npreceq</code>	$\nsucceq$	<code>\nsucceq</code>	$\nVdash$	<code>\nVdash</code>	$\nVDash$	<code>\nVDash</code>
$\nprecneqq$	<code>\nprecneqq</code>	$\succneqq$	<code>\succneqq</code>	$\ntriangleleft$	<code>\ntriangleleft</code>	$\ntriangleright$	<code>\ntriangleright</code>
$\nprecnsim$	<code>\nprecnsim</code>	$\succnsim$	<code>\succnsim</code>	$\ntrianglelefteq$	<code>\ntrianglelefteq</code>	$\ntrianglerighteq$	<code>\ntrianglerighteq</code>
$\nprecnapprox$	<code>\nprecnapprox</code>	$\succnapprox$	<code>\succnapprox</code>				

Binary Operators

Standard Operators

$+$	<code>+</code>	$-$	<code>-</code>	$\vee$	<code>\lor</code> or <code>\vee</code>	$\wedge$	<code>\land</code> or <code>\wedge</code>	$\triangleleft$	<code>\lhd</code>	$\triangleright$	<code>\rhd</code>
$\pm$	<code>\pm</code>	$\mp$	<code>\mp</code>	$\oplus$	<code>\oplus</code>	$\ominus$	<code>\ominus</code>	$\trianglelefteq$	<code>\unlhd</code>	$\trianglerighteq$	<code>\unrhd</code>
$\times$	<code>\times</code>	$\cdot$	<code>\cdot</code>	$\odot$	<code>\odot</code>	$\oslash$	<code>\oslash</code>	$\bullet$	<code>\bullet</code>	$\circ$	<code>\circ</code>
$\div$	<code>\div</code>	$\setminus$	<code>\setminus</code>	$\otimes$	<code>\otimes</code>	$\bigcirc$	<code>\bigcirc</code>	$*$	<code>\ast</code>	$\star$	<code>\star</code>
$\cup$	<code>\cup</code>	$\cap$	<code>\cap</code>	$\triangle$	<code>\bigtriangleup</code>	$\nabla$	<code>\bigtriangledown</code>	$\diamond$	<code>\diamond</code>	$\wr$	<code>\wr</code>
$\sqcup$	<code>\sqcup</code>	$\sqcap$	<code>\sqcap</code>	$\triangleleft$	<code>\triangleleft</code>	$\triangleright$	<code>\triangleright</code>	$\amalg$	<code>\amalg</code>	$\uplus$	<code>\uplus</code>

Additional Operators

$\dotplus$	<code>\dotplus</code>	$\centerdot$	<code>\centerdot</code>	$\boxplus$	<code>\boxplus</code>	$\boxminus$	<code>\boxminus</code>	$\divideontimes$	<code>\divideontimes</code>
$\ltimes$	<code>\ltimes</code>	$\rtimes$	<code>\rtimes</code>	$\boxtimes$	<code>\boxtimes</code>	$\boxdot$	<code>\boxdot</code>	$\circledast$	<code>\circledast</code>
$\Cup$ or $\doublecup$	<code>\Cup</code> or <code>\doublecup</code>	$\Cap$ or $\doublecap$	<code>\Cap</code> or <code>\doublecap</code>	$\leftthreetimes$	<code>\leftthreetimes</code>	$\rightthreetimes$	<code>\rightthreetimes</code>	$\circledcirc$	<code>\circledcirc</code>
$\veebar$	<code>\veebar</code>	$\barwedge$	<code>\barwedge</code>	$\curlyvee$	<code>\curlyvee</code>	$\curlywedge$	<code>\curlywedge</code>	$\circledddash$	<code>\circledddash</code>
$\doublebarwedge$	<code>\doublebarwedge</code>	$\intercal$	<code>\intercal</code>	$\setminus$	<code>\setminus</code>	$\smallsetminus$	<code>\smallsetminus</code>		

Large Operators

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

Functions

$\arccos$	<code>\arccos</code>	$\csc$	<code>\csc</code>	$\operatorname{inj\,lim}$	<code>\injlim</code>	$\max$	<code>\max</code>	$\tan$	<code>\tan</code>
$\arcsin$	<code>\arcsin</code>	$\deg$	<code>\deg</code>	$\ker$	<code>\ker</code>	$\min$	<code>\min</code>	$\tanh$	<code>\tanh</code>
$\arctan$	<code>\arctan</code>	$\det$	<code>\det</code>	$\lg$	<code>\lg</code>	$\Pr$	<code>\Pr</code>	$\varinjlim$	<code>\varinjlim</code>
$\arg$	<code>\arg</code>	$\dim$	<code>\dim</code>	$\lim$	<code>\lim</code>	$\operatorname{proj\,lim}$	<code>\projlim</code>	$\varinjlim$	<code>\varinjlim</code>
$\cos$	<code>\cos</code>	$\exp$	<code>\exp</code>	$\liminf$	<code>\liminf</code>	$\sec$	<code>\sec</code>	$\varliminf$	<code>\varliminf</code>
$\cosh$	<code>\cosh</code>	$\gcd$	<code>\gcd</code>	$\limsup$	<code>\limsup</code>	$\sin$	<code>\sin</code>	$\varlimsup$	<code>\varlimsup</code>
$\cot$	<code>\cot</code>	$\hom$	<code>\hom</code>	$\ln$	<code>\ln</code>	$\sinh$	<code>\sinh</code>	$226_0^1$	<code>\operatorname{226}_0^1</code>
$\coth$	<code>\coth</code>	$\inf$	<code>\inf</code>	$\log$	<code>\log</code>	$\sup$	<code>\sup</code>		

Delimiters

Note that you can produce according relatively sized symbols by preceding the commands with `\left` or `\right`. For example, `\left(\frac{abc}{xyz}\right)` turns  $(\frac{abc}{xyz})$  to  $\left(\frac{abc}{xyz}\right)$ . Sometimes commands can be preceded with ‘l’ or ‘r’ e.g., `\lVert xyz\rVert` makes  $\|xyz\|$ . Thus, giving the `\Vert` command properties of paired symbols.

Standard Delimiters

(	(	[	\lbrack or [	<	\langle	⌊	\lfloor	⌞	\ulcorner	↑	\uparrow
)	)	]	\rbrack or ]	>	\rangle	⌋	\lceil	⌟	\urcorner	↓	\downarrow
	\vert or	{	\lbrace or \{	⌈	\lceil	/	/	⌘	\llcorner	↗	\Uparrow
	\Vert or \	}	\rbrace or \}	⌋	\lfloor	\	\backslash	⌞	\rcorner	↘	\Downarrow

Large Delimiters

(	\lgroup	)	\rgroup	⸮	\lmoustache	⸮	\rmoustache		\arrowvert		\Arrowvert	,	\bracevert
---	---------	---	---------	---	-------------	---	-------------	--	------------	--	------------	---	------------

Arrows

Standard Arrows

←	\leftarrow or \gets	→	\rightarrow or \to	⇐	\Leftarrow	⇒	\Rightarrow
⇐	\longleftarrow	→	\longrightarrow	⇐	\Longleftarrow	⇒	\Longrightarrow
↔	\leftrightharrow	↔	\longlleftrightharrow	↔	\Leftrightharrow	↔	\Longleftrightharrow
↑	\uparrow	↓	\downarrow	↗	\Uparrow	↘	\Downarrow
↕	\updownarrow	↦	\mapsto	↕	Updownarrow	↦	\longmapsto
↩	\hookleftarrow	↪	\hookrightarrow	⇔	\iff (larger spaces)		
↗	\nearrow	↘	\searrow	↗	\swarrow	↖	\nwarrow
↤	\nleftarrow	↦	\nrightarrow	↤	\nLeftarrow	↦	\nRightarrow
↔	\nleftrightharrow			↔	\nLeftrightharrow		

Special Arrows

↔	\dashleftarrow	→	\dashrightarrow	⇔	\leftleftarrows	⇒	\righttroghtarrows	⇔	\leftrightharrows
⇐	\Lleftarrow	⇒	\Rrightarrow	↗	\upuparrows	↘	\downownarrows	⇔	\rightleftarrows
↑	\upharpoonleft	↑	\upharpoonright	↓	\downharpoonleft	↓	\downharpoonright	⇐	\leftrightharpoons
↔	\twoheadleftarrow	→	\twoheadrightarrow	↔	\leftarrowtail	→	\rightarrowtail	⇐	\rightleftharpoons
↖	\Lsh	↗	\Rsh	↗	\looparrowleft	↖	\looparrowright		
↶	\curvearrowleft	↷	\curvearrowright	↺	\circlearrowleft	↻	\circlearrowright		
↪	\leadsto	↪	\rightsquigarrow	↪	\leftrightsquigarrow	↪	\multimap		

Matrices & Arrays

Note that any of the following can also be displayed inline as well as stand-alone. It’s recommended that you use `smallmatrix` for this. Thus, you must preceed and succeed `\begin` and `\end smallmatrix` with `\left<delimiter>` and `\right<delimiter>`, respectively. For example, `\left(\begin{smallmatrix}a & b & c\\x & y & z\end{smallmatrix}\right)` yields  $\left(\begin{smallmatrix}a & b & c\\x & y & z\end{smallmatrix}\right)$ .

Basic Syntax

$\begin{matrix} a & b & c \\ x & y & z \end{matrix}$	<code>\begin{matrix}</code> <code>a &amp; b &amp; c\\</code> <code>x &amp; y &amp; z</code> <code>\end{matrix}</code>	$\begin{pmatrix} a & b & c \\ x & y & z \end{pmatrix}$	<code>\begin{pmatrix}</code> <code>a &amp; b &amp; c\\</code> <code>x &amp; y &amp; z</code> <code>\end{pmatrix}</code>	$\begin{bmatrix} a & b & c \\ x & y & z \end{bmatrix}$	<code>\begin{bmatrix}</code> <code>a &amp; b &amp; c\\</code> <code>x &amp; y &amp; z</code> <code>\end{bmatrix}</code>
$\left\{ \begin{matrix} a & b & c \\ x & y & z \end{matrix} \right\}$	<code>\begin{Bmatrix}</code> <code>a &amp; b &amp; c\\</code> <code>x &amp; y &amp; z</code> <code>\end{Bmatrix}</code>	$\begin{vmatrix} a & b & c \\ x & y & z \end{vmatrix}$	<code>\begin{vmatrix}</code> <code>a &amp; b &amp; c\\</code> <code>x &amp; y &amp; z</code> <code>\end{vmatrix}</code>	$\left\  \begin{matrix} a & b & c \\ x & y & z \end{matrix} \right\ $	<code>\begin{Vmatrix}</code> <code>a &amp; b &amp; c\\</code> <code>x &amp; y &amp; z</code> <code>\end{Vmatrix}</code>
$\left[ \begin{matrix} a & b & c \\ x & y & z \end{matrix} \right]$	<code>\left\lceil\begin{matrix}</code> <code>a &amp; b &amp; c\\</code> <code>x &amp; y &amp; z</code> <code>\right\rceil\end{matrix}</code>	$\left\langle \begin{matrix} a & b & c \\ x & y & z \end{matrix} \right\rangle$	<code>\left\lceil\begin{matrix}</code> <code>a &amp; b &amp; c\\</code> <code>x &amp; y &amp; z</code> <code>\right\rceil\end{matrix}</code>	<code>\left\langle\begin{matrix}</code> <code>a &amp; b &amp; c\\</code> <code>x &amp; y &amp; z</code> <code>\right\rceil\end{matrix}</code>	

Dots

...	\dots or \ldots	...	\cdots	...	\ddots	:	\vdots
-----	-----------------	-----	--------	-----	--------	---	--------

Array Environment

Note that arrays operate in the same manner as tables such that they permit column alignment l, c and r etc., columns can be divided using pipes (|) new row lines with \\, and the use of `\hline`, to name a few examples. Columns are separated the same as within tables; with  $(n - 1)$  & ampersand symbols, for  $n$  columns. Some simple examples follow.

$\begin{array}{lcr} a & b & c \\ x & y & z \\ k & j & i \end{array}$	<code>\begin{array}{lcr}</code> <code>a &amp; b &amp; c\\</code> <code>x &amp; y &amp; z\\</code> <code>k &amp; j &amp; i</code> <code>\end{array}</code>	$\begin{array}{lcr} \frac{a}{x} & \frac{b}{y} & \frac{c}{z} \\ a & b & c \\ k & j & i \end{array}$	<code>\begin{array}{lcr}</code> <code>a &amp; b &amp; c\\</code> <code>x &amp; y &amp; z\\</code> <code>k &amp; j &amp; i</code> <code>\end{array}</code>
--	---	--	---

