

Symbols & Logical Syntax in L^AT_EX

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, ς	<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, χ	<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

<i>F</i>	<code>\digamma</code>	℄	<code>\complement</code>	∠	<code>\angle</code>	ℑ	<code>\Im</code>	⊃	<code>\Game</code>	£	<code>\pounds</code>
ℵ	<code>\aleph</code>	ℓ	<code>\ell</code>	∠	<code>\measuredangle</code>	ℜ	<code>\Re</code>	⊄	<code>\Finv</code>	\$	<code>\\$</code>
⊔	<code>\beth</code>	℘	<code>\eth</code>	∠	<code>\sphericalangle</code>	ℳ	<code>\mho</code>	∂	<code>\partial</code>	?	<code>?</code>
⌈	<code>\daleth</code>	ℏ	<code>\hbar</code>	√	<code>\surd</code>	℘	<code>\wp</code>	TM , ©	<code>\trademark, \copyright</code>	!	<code>!</code>
⌋	<code>\gimel</code>	ℓ	<code>\hslash</code>	ℵ	<code>\natural</code>	ℓ	<code>\Bbbk</code>	®	<code>\textregistered, \circledR</code>	.	<code>.</code>
ι	<code>\imath</code>	⊤	<code>\top</code>	♯	<code>\sharp</code>	∅	<code>\emptyset</code>	◇, ◇	<code>\diamondsuit, \lozenge</code>	,	<code>,</code>
ⵍ	<code>\jmath</code>	⊥	<code>\bot</code>	♭	<code>\flat</code>	∞	<code>\infty</code>	♥	<code>\heartsuit</code>	‘, ’	<code>‘, ’</code>
∇	<code>\nabla</code>	§	<code>\S</code>	Δ	<code>\vartriangle</code>	□, □	<code>\Box, \square</code>	♣	<code>\clubsuit</code>	“, ”	<code>‘, ’, ’ or "</code>
△	<code>\triangle</code>	∅	<code>\varnothing</code>	▽	<code>\triangledown</code>	◇	<code>\Diamond</code>	♠	<code>\spadesuit</code>	∴, ∵	<code>\therefore, \because</code>
▲	<code>\blacktriangle</code>	■	<code>\blacksquare</code>	↘	<code>\diagdown</code>	∃	<code>\exists</code>	★	<code>\bigstar</code>		
▼	<code>\blacktriangledown</code>	◆	<code>\blacklozenge</code>	↗	<code>\diagup</code>	∄	<code>\nexists</code>				

Basic Math Mode

Math Alphabets

<i>XYZ xyz</i>	<code>XYZ\ xyz</code>	<i>XYZ xyz</i>	<code>\mathrm{XYZ\ xyz}</code>	<i>XYZ xyz</i>	<code>\mathit{XYZ\ xyz}</code>	XYZ xyz	<code>\mathbf{XYZ\ xyz}</code>
XYZ	<code>\mathbb{XYZ}</code>	ℳ	<code>\mathcal{XYZ}</code>	Ⅎ	<code>\mathfrak{XYZ\ xyz}</code>	XYZ	<code>\mathtt{XYZ\ xyz}</code>
XYZ xyz	<code>\mathsf{XYZ\ xyz}</code>						

Math Spacing

<i>xyz</i>	<i>xyz</i>	Math spacing	<i>sin x cos y</i>	<code>\sin x\cos y</code>	Operator spacing
<i>x y z</i>	<code>x\ y\ z</code>	Extended spacing	<i>a b c d</i>	<code>a\,b\mspace{3mu}c\thinspace d</code>	3mu (‘thin’) space
<i>a b c d</i>	<code>a\b\mspace{4mu}c\medspace d</code>	4mu (‘medium’) space	<i>a b c d</i>	<code>a\;b\mspace{5mu}c\thickspace d</code>	5mu (‘thick’) space
<i>a b c d</i>	<code>a\quad b\mspace{18mu}c\quad d</code>	18mu (‘quad’) space	<i>d x l</i>	<code>a\!b\mspace{-3mu}c\negthinspace d</code>	Neg. 3mu (‘thin’) space
<i>a b</i>	<code>ab</code>	Space width of ‘xxx’			

Math Accents & Constructs

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

$<$	<code><</code>	$>$	<code>></code>	$=$	<code>=</code>	\in	<code>\in</code>	\ni or \owns	<code>\ni or \owns</code>
\leq	<code>\leq or \le</code>	\geq	<code>\geq or \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>	\nexists or $\neg\exists$	<code>\nexists or \neg\exists</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>
\sqsubsetseteq	<code>\sqsubsetseteq</code>	\sqsupsetseteq	<code>\sqsupsetseteq</code>	\bowtie	<code>\bowtie</code>	\notin	<code>\notin</code>	\prime, \backprime	<code>\prime, \backprime</code>

Binary Operators

Standard Operators													
$+$	<code>\pm</code>	$-$	<code>\mp</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>	\dagger	<code>\dagger</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>	\ddagger	<code>\ddagger</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>	\P	<code>\P</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>	\triangleup	<code>\bigtriangleup</code>	∇	<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>		

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>	\iiiiiint	<code>\iiiiiint</code>	\bigcup	<code>\bigcup</code>	\bigotimes	<code>\bigotimes</code>	\bigwedge	<code>\bigwedge</code>
\coprod	<code>\coprod</code>	\iint	<code>\iint</code>			\biguplus	<code>\biguplus</code>	\bigodot	<code>\bigodot</code>	\bigsqcup	<code>\bigsqcup</code>

Functions

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

$($	$)$	$[$	$]$	\langle	\rangle	\lfloor	\lceil	\ulcorner	\urcorner	\llcorner	\lrcorner	\uparrow	\downarrow	\Uparrow	\Downarrow
\lvert	\rvert	$\{$	$\}$	\lceil	\rceil	\lfloor	\rfloor	\ulcorner	\urcorner	\llcorner	\lrcorner	\uparrow	\downarrow	\Uparrow	\Downarrow

Large Delimiters

\langle	\rangle	\lgroup	\rgroup	\loustache	\rmoustache	\uparrow	\downarrow	\Uparrow	\Downarrow
-----------	-----------	-----------	-----------	--------------	---------------	------------	--------------	------------	--------------

Arrows

\leftarrow	\longleftarrow	\leftrightarrow	\uparrow	\Uparrow	\hookrightarrow	\nearrow	\rightarrow	\longrightarrow	\rightarrowtail	\curvearrowright	\searrow	\Leftarrow	\Longleftarrow	\Leftrightarrow	\Uparrow	\Updownarrow	\iff (larger spaces)	\swarrow	\Rightarrow	\Longrightarrow	\Rightarrowtail	\looparrowright	\circlearrowleft	\multimap
\dashleftarrow	\Lleftarrow	\upharpoonleft	\twoheadleftarrow	\Lsh	\curvearrowleft	\leadsto	\dashrightarrow	\Rrightarrow	\rightarrowtail	\curvearrowright	\searrow	\Leftarrow	\Longleftarrow	\Leftrightarrow	\Uparrow	\Updownarrow	\iff (larger spaces)	\swarrow	\Rightarrow	\Longrightarrow	\Rightarrowtail	\looparrowright	\circlearrowleft	\multimap

...

Matrices

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 $\left(\begin{smallmatrix}a & b & c \\ x & y & z\end{smallmatrix}\right)$.

Syntax

$\begin{matrix} a & b & c \\ x & y & z \end{matrix}$	$\begin{matrix} a & b & c \\ x & y & z \end{matrix}$	$\begin{pmatrix} a & b & c \\ x & y & z \end{pmatrix}$	$\begin{bmatrix} a & b & c \\ x & y & z \end{bmatrix}$	$\begin{matrix} a & b & c \\ x & y & z \end{matrix}$
$\begin{Bmatrix} a & b & c \\ x & y & z \end{Bmatrix}$	$\begin{bmatrix} a & b & c \\ x & y & z \end{bmatrix}$	$\begin{vmatrix} a & b & c \\ x & y & z \end{vmatrix}$	$\begin{Vmatrix} a & b & c \\ x & y & z \end{Vmatrix}$	$\begin{matrix} a & b & c \\ x & y & z \end{matrix}$

$$\begin{bmatrix} a & b & c \\ x & y & z \end{bmatrix}$$

```
\left\lceil\begin{matrix}
a & b & c\\
x & y & z
\end{matrix}\right\rceil
```

$$\left\langle \begin{matrix} a & b & c \\ x & y & z \end{matrix} \right\rangle$$

```
\left\langle\begin{matrix}
a & b & c\\
x & y & z
\end{matrix}\right\rangle
```

Dots

$$\dots$$

```
\dots or \ldots
```

$$\cdots$$

```
\cdots
```

$$\ddots$$

```
\ddots
```

$$\vdots$$

```
\vdots
```