

Math in L^AT_EX

Eric Eaton

September 7, 2004

1 General Information

Special Characters: # \$ % & { } - ~ ^ \. The first seven can be generated with \ followed by the character.

Quotation marks: ‘‘\ldots’’ yields “...”.

Dashes: - (hyphen -), -- (range -), and --- (dash —).

Spaces between words don't matter a bit. Forced hyphenation can be done this way: `hy\phen\-a\-tion`.

`\emph{Emphasized Text}` = *Emphasized Text*

`{\bf Bold Text}` = **Bold Text**

`{\tt Typewriter Text}` = Typewriter Text

To prevent line breaks between certain words, use a ~ in place of a space: Mr. Johns, Figure 7 (Mr.~Johns, Figure~7).

To print long text on a single line, surround it by an \mbox: `\mbox{This text will appear on one line.}`

Footnotes¹ can be printed using `\footnote{This is a footnote}`.

Comments are done by % characters.

Vertical space can be specified by `\vspace{0.25in}`

Verbatim text can be done with `\verb+Text!+` or the verbatim environment `\begin{verbatim} ... \end{verbatim}`.

¹This is a footnote.

Lists types: itemize, enumerate, and description.

```
\begin{itemize}
  \item \begin{description}
    \item[Name1] Description of the Name1 object.
    \item[Name2] Description of the Name2 object.
  \end{description}
  \item \begin{enumerate}
    \item item1
    \item item2
  \end{enumerate}
\end{itemize}
```

- **Name1** Description of the Name1 object.
Name2 Description of the Name2 object.
- 1. item1
2. item2

2 Types of Math Environments

Formulas in the middle of text: $(2x=4)$ or $\$2x=4\$$. This appears as follows: $2x = 4$ or $2x = 4$. `\begin{math} \end{math}` also works the same.

The `displaymath` environment offsets equations for emphasis:

```
\begin{displaymath}
  2x=4
\end{displaymath}
```

$$2x = 4$$

The commands `\[\]` are shortcuts to the `displaymath` environment.

The `equation` environment does the same, but numbers the equation:

```
\begin{equation}
  2x=4
  \label{simpleequation}
\end{equation}
```

$$2x = 4 \tag{1}$$

To refer to Equation 1 (`Equation~\ref{simpleequation}`), use a cross-reference. Note that this takes multiple passes of \LaTeX . In figures, the label command must go after the caption command.

3 Typical Math Constructs

Exponentiation: $x^2 = x^2$, $x^{2y} = x^{2y}$, $x^{4^y} = x^{4^y}$.

Subscripts: $x_2 = x_2$, $x^{y_2} = x^{y_2}$.

Fractions: $n/(2+m) = n/(2+m)$, $\left[\frac{y+z/2}{y^2+1} \right] =$

$$\frac{y + z/2}{y^2 + 1}$$

Ellipsis: $\text{\ldots} = \dots$

Roots: $\sqrt{x+y} = \sqrt{x+y}$, $\sqrt[n]{x+y} = \sqrt[n]{x+y}$

Greek letters: $\alpha = \alpha$, $\beta = \beta$, $\delta = \delta$, $\Delta = \Delta$, $\theta = \theta$, $\pi = \pi$, ...

Common operators:

\times	\div	\pm	\bullet
\cap	\cup	\subset	\supset
\vee	\wedge	\neg	\in
\leq	\geq	\neq	\equiv
∞	\forall	\exists	\emptyset
\leftarrow	\Leftrightarrow	\leftrightarrow	\Leftrightarrow

\times	\div	\pm	\bullet
\cap	\cup	\subset	\supset
\vee	\wedge	\neg	\in
\leq	\geq	\neq	\equiv
∞	\forall	\exists	\emptyset
\leftarrow	\Leftrightarrow	\leftrightarrow	\Leftrightarrow

3.1 Functions

\sum	\prod	\int
\bigcup	\bigcap	\oint

\sum	\prod	f
\cup	\cap	f

These functions all work similar to the following example:

$\sum_{i=0}^n x_i = \int_0^1 f$. This expression will look this way $\sum_{i=0}^n x_i = \int_0^1 f$ when in the text but this way

$$\sum_{i=0}^n x_i = \int_0^1 f$$

when in displaymath mode.

Common math functions:

```
\log      \cos      \sin      \tan
\arccos   \arcsin   \cosh     \sinh
\lim      \ln        \max      \min
    log      cos      sin      tan
    arccos   arcsin   cosh     sinh
    lim      ln       max      min
```

Example with limits:

```
\lim_{n \rightarrow \infty} x = 0
```

$\lim_{n \rightarrow \infty} x = 0$ and in displaymath mode:

$$\lim_{n \rightarrow \infty} x = 0$$

Picky things about functions:

- To typeset $\{a \mid a > 0\}$, try $\{a \mid a > 0\}$. The point is to use `\mid` instead of `|`. The latter has spacing problems.
- To typeset $f: X \rightarrow Y$, try $f: X \rightarrow Y$. Use `\colon` instead of `:` to get the correct spacing.
- To typeset multi-character names in math mode, use `\mathit`. For example, doesn't *difference* = 1 look better than *difference* = 1? The commands `\textstyle` (for in-text math) and `\displaystyle` (for displaymath mode) can be used to make plain text in an equation: `\[\displaystyle{Let\,} x=1.\]`

$$Let x = 1.$$

Notice the extra space by `\,`! You can also use `\mbox` to make plain text.

You can define your own function in the following manner:

```
\newcommand{\SumToX}[2]{\ensuremath{\sum_{\#1=1}^{\#2}}}
\newcommand{\QuadraticFormula}[3]{\ensuremath{\frac{-\#2 \pm
\sqrt{\#2^2 - 4 \times \#1 \times \#3}}{2 \times \#1}}}
```

Then we can use the new commands in our document: $\$\text{\SumToX{i}{N}x_i}\$ = \sum_{i=1}^N x_i$.
`\[\text{\QuadraticFormula{a}{b}{c}}\]` =

$$\frac{-b \pm \sqrt{b^2 - 4 \times a \times c}}{2 \times a}$$

3.2 Arrays

Note that the `\tabular` environment is very similar to the `\array` environment, except it is for regular text.

```
\( \begin{array}{c|l|r}
Name1 & Name2 & Name3 \\ \hline
a & xy & 12 \\
a+b & x+y & 5 \\
a+b+c & x/y & 100
\end{array} \)
```

<i>Name1</i>	<i>Name2</i>	<i>Name3</i>
<i>a</i>	<i>xy</i>	12
<i>a + b</i>	<i>x + y</i>	5
<i>a + b + c</i>	<i>x/y</i>	100

Delimiters are often used in combination with arrays. The delimiters automatically scale to encompass the arrays. Use the commands `\left` or `\right` before a delimiter to specify the left or right side. Common delimiters:

```
( ) [ ]
\{ \} | \|
\lfloor \rfloor \lceil \rceil

( ) [ ]
{ } | ||
[ ] [ ]
```

An example:

```
\[ \left( \begin{array}{c}
\left[ \begin{array}{cc} x_1 & x_2 \\ x_3 & x_4 \end{array} \right] \\
y \\
z
\end{array} \right) \]
```

$$\left(\begin{array}{c} \left[\begin{array}{cc} x_1 & x_2 \\ x_3 & x_4 \end{array} \right] \\ y \\ z \end{array} \right)$$

The argument t aligns the top line of the second array with the center of the first. The argument b would align the bottom line with the center.

```
\[ X = \left[ \begin{array}{c} a_1 \\ \dots \\ a_n \end{array} \right] - \\
\left[ \begin{array}{t}{cc} x-y & x+y \\ xy & x/y \end{array} \right] \]
```

$$X = \begin{bmatrix} a_1 \\ \dots \\ a_n \end{bmatrix} - \begin{matrix} x - y & x + y \\ xy & x/y \end{matrix}$$

You can make an invisible delimiter with a “.” as follows:

```
\[x = \left\{ \begin{array}{l}
y & \mbox{if } \$y>0\$ \\
0 & \mbox{otherwise}
\end{array} \right. . \]
```

$$x = \begin{cases} y & \text{if } y > 0 \\ 0 & \text{otherwise} \end{cases}$$

3.3 Equation Arrays

Equation arrays allow you to create an aligned series of equations. Each equation can either be numbered (using `\eqnarray`) or unnumbered (using `\eqnarray*`). A `\nonumber` command on a line tells L^AT_EX to not number that line. Here are two examples:

```
\begin{eqnarray}
x & = & 5y + 6z \\
y & > & a + b + c + d + \\
& & e + f + g \nonumber
\end{eqnarray}
```

```
\begin{eqnarray*}
10 & = & 5x \\
x & = & 10/5 \\
x & = & 2
\end{eqnarray*}
```

$$x = 5y + 6z \tag{2}$$

$$y > a + b + c + d + e + f + g \tag{3}$$

$$10 = 2x + 3x$$

$$10 = x(2 + 3)$$

$$x = 2$$

3.4 Stacking

You can overline with the `\overline` command and underline with the `\underline` command. For example, `\overline{\overline{y}^3 + 1} = \underline{3x}` yields $\overline{\overline{y}^3 + 1} = \underline{3x}$.

Overbracing and underbracing works similarly: `\overbrace{w + \underbrace{x + y}_{12}}^{24}` yields $\overbrace{w + \underbrace{x + y}_{12}}^{24}$.

Some common math accents:

```
(\begin{array}{cccc}
\hat{x} & \bar{x} & \vec{x} & \dot{x} \\
\end{array} \ )
```

\hat{x} \bar{x} \vec{x} \dot{x}

The letters i and j should not have dots when accented, so use `\imath` and `\jmath` to produce these: $\vec{i} + \vec{j}$.

The `\stackrel` command allows us to stack arbitrary symbols:

`\vec{X} \stackrel{\mathit{def}}{=} (x_1, \dots, x_n)` yields $\vec{X} \stackrel{\mathit{def}}{=} (x_1, \dots, x_n)$.

3.5 Theorems and Such

We can define and automatically number theorems as shown in the following examples:

```
\newtheorem{theorem}{Theorem}
\newtheorem{axiom}{Axiom}
```

```
\begin{theorem}
  This is a theorem.
  \label{TheoremThis}
\end{theorem}
```

```
\begin{axiom}
  All theorems are dull.
  \label{AxiomDullTheorems}
\end{axiom}
```

Theorem 1 *This is a theorem.*

Axiom 1 *All theorems are dull.*

By Axiom 1, we can state that Theorem 1 is dull.