

Getting Started with L^AT_EX

Rebekah B. Johnson Yates
Department of Mathematics and Computer Science
Houghton College, Houghton, NY

August 26, 2010; modified January 19, 2013

1 Getting Started

1.1 Step 1

You need to download a T_EX distribution. If you are using Windows, the best place to do this is <http://miktex.org/>. Click on MiKTeX 2.9 under Download in the left menu bar. Then under **Installing a basic MiKTeX system**, choose one of the options (it's probably best to choose one from the United States) from the from: drop-down menu and click the Download button. Open the .exe file, accept the terms, and keep hitting next until you get to the page asking you to choose paper size. For paper size, choose letter, and choose yes for "install packages on-the-fly" (you will not install anything that will harm your computer).

If you have a Mac, you can download MacT_EX from <http://www.tug.org/mactex>. You may also be able to use MiKTeX. Just follow the directions on that webpage.

If you use Linux, use your package manager to get T_EX and then use your favorite text editor.

1.2 Step 2

While you can write in L^AT_EX in notepad or any text editor and compile your code using the command window, it is generally easier and faster, to use some sort of editor created for L^AT_EX. If you are using Windows, I recommend the free, open source editor T_EXnicCenter, which you can download from <http://www.texniccenter.org/>. Follow the directions on the website and install the editor.

If you have a Mac, you can use the free, open source editor T_EXShop, which is available at

<http://www.uoregon.edu/~koch/texshop/index.html>.

1.3 Step 3

For T_EXnicCenter users, when you open T_EXnicCenter for the first time, you will most likely need to tell the program where to find MiKTeX. Go through the steps in the Start Up wizard; when you get to the step where it asks for your L^AT_EX distribution, find the folder where you have MiKTeX; in that folder, open the miktex folder; in the miktex folder, you should have another folder labeled bin. That is the folder to which you need to direct T_EXnicCenter. If you're unsure about this, bring your computer to my office. Whatever you do, do not skip the startup wizard!

If you're using T_EXShop, search for help online (I can try to help you, but I may not be terribly successful).

2 Setting up your document

The first thing you need to have in your text file is the document setup. At the learning stage, I would recommend just copying the setup from the beginning of the L^AT_EX files I give you. In the future, you may choose to change these however you want, but at this point you do not want to get bogged down; you want to learn how to write L^AT_EX!

2.1 Preamble

The basic things you might want to know about the document preamble are that you need a document class (for homework and papers, the article class will be what you use), followed by packages you may use in your document. The preamble will look something like this:

```
\documentclass[11pt]{article}
\usepackage{amsmath, amssymb, amscd, amsthm, mathrsfs, url}
\usepackage{fancyhdr}
\usepackage[left=.5in,top=.75in,right=.5in,bottom=.75in]{geometry}

\pagestyle{fancy}
\lhead{Getting Started with \LaTeX{}}
\rhead{Rebekah Yates}
```

The font size after document class is in square brackets because it is an optional parameter; if you do not put it there, L^AT_EX will use the default size for the document class you have chosen.

The packages you're telling L^AT_EX to use contain several symbols that we will use; it is not important at this stage to know what all of them do, nor do you necessarily need all of these (though you do need some of these specific ones), but you may as well keep the ones I have in example files. If you need a symbol which is not in one of these packages, you can find the symbol in the *The Comprehensive List of L^AT_EX Symbols* where you will also find which package you need to have for that. T_EXnicCenter will find and install packages for you from reputable sources on the internet, but it will warn you first that it is doing so.

2.2 Body

Now that you have the preamble covered, you are ready to write the actual content of your document. You need to tell L^AT_EX where the content begins and where it ends, so you place this after the preamble:

```
\begin{document}
Your wonderful and amazing thoughts that you want to typeset beautifully go here.
\end{document}
```

3 What Goes in the Body

3.1 L^AT_EX is not a word processor

You are writing code in an ASCII environment that compiles externally to a .pdf or .dvi file. Writing in L^AT_EX is much more similar to programming than it is to creating a document using a word processor. You give L^AT_EX the content; L^AT_EX does the typesetting work for you.

3.1.1 Spaces

White spaces in your L^AT_EX code are not interpreted in the same way as white spaces in a word processing environment. Spaces in the same line are interpreted as one space, e.g.

```
No matter   how many           spaces   I put between these words, they will   be
the same                                     space apart when I compile.
```

renders as

No matter how many spaces I put between these words, they will be the same space apart when I compile.

Hitting enter at the end of a line in your L^AT_EX code does not make it compile the text on separate lines; such lines are all in one paragraph. You must put a vertical space between lines in order to have the text on separate lines (a vertical space signals the beginning of a new paragraph):

```
These two lines of code
will produce output on the same line.
```

```
But this line, preceded by a vertical space, will begin on a new line.
```

becomes

These two lines of code will produce output on the same line.

But this line, preceded by a vertical space, will begin on a new line.

Spacing between paragraphs is universal to your document (you can set it in the preamble), so more than one vertical space between paragraphs in your code will not change the amount of space between paragraphs in the output.

3.2 Mathematical Content

For us, the main point of learning to use L^AT_EX is to be able to beautifully typeset mathematical notation. For a more comprehensive explanation of why one should choose to use L^AT_EX for crafting documents, especially mathematical documents, see *The Not So Short Introduction to L^AT_EX 2 ϵ* , Chapter 1 (this entire document is an extremely helpful reference in learning and using L^AT_EX).

3.2.1 Math Mode

In order to type mathematical symbols, equations, etc., in your document, you must warn L^AT_EX that you are using mathematical notation by telling it to read the notation in math mode. There are several ways to signal math mode.

First, you can put things in math mode by surrounding them with dollar signs:

```
$f(x) = 2x - 3$
```

renders as $f(x) = 2x - 3$. Doing this will keep your mathematical notation inline with the rest of your writing.

Sometimes, you want your equations to be set apart from the rest of the text on their own line. You can do this by using the following form of math mode:

```
\begin{equation}
e^{i\pi} = -1.
\end{equation}
```

renders as

$$e^{i\pi} = -1. \tag{1}$$

When you compile your document, you will notice that this equation is numbered. If you would rather not have your equation numbered (the reason for this is generally that you will not need to refer to the specific equation later in your document), use this form:

```
\begin{equation*}
\cos^2 \theta + \sin^2 \theta = 1,
\end{equation*}
```

which renders as

$$\cos^2 \theta + \sin^2 \theta = 1,$$

an unnumbered equation. Notice that using an equation environment sets apart and centers your equations/expressions. This often makes a paper much more readable than using inline mathematics.

A third form of math mode is useful when you want to have more than one line of math set apart from the rest of the text. The commands

```
\begin{align}
\sum_{i = 1}^n i \quad &= \frac{n(n+1)}{2} \\
\sum_{i = 1}^n i^2 \quad &= \frac{n(n+1)(2n+1)}{6}
\end{align}
```

will compile as

$$\sum_{i=1}^n i = \frac{n(n+1)}{2} \tag{2}$$

$$\sum_{i=1}^n i^2 = \frac{n(n+1)(2n+1)}{6} \tag{3}$$

Notice that the comment in the second line of the L^AT_EX code (the phrase that is preceded by a %) does not appear when the code is compiled.

Of course, in this non-asterisked align environment, each of the lines are numbered when you compile the document. If you have no need to refer to the individual lines later in your document, you can use this form:

```
\begin{align*}
\frac{d}{dx} x^3 \quad &= \lim_{h \to 0} \frac{(x+h)^3 - x^3}{h} \\
&= \lim_{h \to 0} \frac{x^3 + 3x^2h + 3xh^2 + h^3 - x^3}{h} \\
&= \lim_{h \to 0} \frac{h(3x^2 + 3xh + h^2)}{h} \\
&= \lim_{h \to 0} (3x^2 + 3xh + h^2) \\
&= 3x^2.
\end{align*}
```

will compile as

$$\begin{aligned} \frac{d}{dx} x^3 &= \lim_{h \rightarrow 0} \frac{(x+h)^3 - x^3}{h} \\ &= \lim_{h \rightarrow 0} \frac{x^3 + 3x^2h + 3xh^2 + h^3 - x^3}{h} \\ &= \lim_{h \rightarrow 0} \frac{h(3x^2 + 3xh + h^2)}{h} \\ &= \lim_{h \rightarrow 0} (3x^2 + 3xh + h^2) \\ &= 3x^2. \end{aligned}$$

3.2.2 Symbols in Math Mode

The commands for many commonly used mathematical symbols are pretty intuitive. All Greek letters are their transliteration, e.g. use

`\alpha`, `\beta`, `\gamma`

to display α , β , γ , etc. If you want a capital Greek letter such as Δ , Γ , or Π , you capitalize the first letter of the command, e.g.,

`\Delta`, `\Gamma`, `\Pi`

etc., for those which are different from our usual alphabet.

When you want to use certain symbols which are also involved in commands, you need to precede the symbol by a backslash, e.g., for `{}`, `#`, `$`, use

`\{ \}`, `\#`, `\$`

Most other commands can be found in either the Math menu in T_EXnic Center or in *The Comprehensive List of L^AT_EX symbols*.

symbols in math mode

symbol	L ^A T _E X command	symbol	L ^A T _E X command
α	<code>\alpha</code>	∞	<code>\infty</code>
β	<code>\beta</code>	$\frac{1}{2}$	<code>\frac{1}{2}</code>
γ	<code>\gamma</code>	x^2	<code>x^2</code>
δ	<code>\delta</code>	$\sum_{n=1}^{\infty} x_n$	<code>\sum_{n = 1}^{\infty} x_n</code>
ϵ	<code>\epsilon</code>	$\lim_{x \rightarrow 1} f(x)$	<code>\lim_{x \to 1} f(x)</code>
θ	<code>\theta</code>	\mathbb{N}	<code>\mathbb{N}</code>
λ	<code>\lambda</code>	\mathbb{Z}	<code>\mathbb{Z}</code>
μ	<code>\mu</code>	\mathbb{Q}	<code>\mathbb{Q}</code>
π	<code>\pi</code>	\mathbb{R}	<code>\mathbb{R}</code>
ρ	<code>\rho</code>	\mathbb{C}	<code>\mathbb{C}</code>
σ	<code>\sigma</code>	$\int_0^{\pi} \sin x dx$	<code>\int_0^{\pi} \sin x dx</code>
τ	<code>\tau</code>	$\sqrt{-1} = i$	<code>\sqrt{-1} = i</code>
ϕ	<code>\phi</code>	$e^{i\theta} = \cos \theta + i \sin \theta$	<code>e^{i\theta} = \cos\theta + i\sin\theta</code>
ψ	<code>\psi</code>	\bar{z}	<code>\overline{z}</code>
Γ	<code>\Gamma</code>	$ x + iy = \sqrt{x^2 + y^2}$	<code> x + iy = \sqrt{x^2 + y^2}</code>
Ω	<code>\Omega</code>	$\{n : n \in \mathbb{N}\}$	<code>\{n:n\in\mathbb{N}\}</code>

4 Compiling

Once you have input (at least some of) your document’s content, you need to compile that document to see your output. If you are using T_EXnicCenter, you can press the compile button in the tool bar (it looks like an arrow pointing down onto a stack of paper). If you would like to view your output as soon as it’s compiled, choose the compile and view button which looks like the other button with the addition of a magnifying glass. If you choose to compile to a .pdf (the choice is in a drop-down menu in your toolbar), you will need a .pdf viewer (such as Acrobat) to look at your output. If you choose to compile to a .dvi, you will need a .dvi viewer (such as YaP—a free viewer that you should be able to find on the internet without much trouble). Other editors should have similar options.

Note: for some reason, Adobe decided to be difficult when writing Acrobat Reader X. To be able to compile and view at once using T_EXnicCenter and Acrobat Reader X, you will need to do a few set up steps manually.

1. First, under the Build menu at the top, go to Define Output Profiles. In the Viewer menu at the right, for Path of Executable, you need to put the executable file where Acrobat Reader is on your computer. For example, mine is

```
C:\PROGRAM FILES\ADOBE\READER 10.0\READER\ACRORD32.EXE
```

Note the difference with Reader 10.0 instead of 9.

2. Next, change all (there are 3) the Server options in the Viewer menu to acroviewR10.
3. Finally, open Adobe Reader X, go to Edit→Preferences→General and uncheck “Enable Protected Mode at startup.” If this still doesn’t work, try opening Adobe Reader X from your start menu and then compiling and viewing from T_EXnicCenter once. After one time, this step should no longer be necessary.

You can also compile your document using the command line. If this is your preference, I assume you know how to use a command line. Use the following command:

```
C:\...\> pdflatex [filename]
```

4.1 Errors

If you have made a mistake in your L^AT_EX code, when you compile your document, you will have errors. It is your job to fix these errors. Your L^AT_EX editor will give you a list of the errors (in T_EXnicCenter, this list appears below your .tex file as the Output Profile), usually with a line number in which they are occurring. Common errors include

- forgetting to put \$ on one or both sides of inline math mode,
- missing an ending } or some other type of bracket, and
- incorrect code structure within a specific environment (such as not having the correct alignment characters in an align environment).

If you are using the command line to compile your document, after an error is reported you must hit Ctrl+C to return to the command prompt.

4.2 Warnings

You may have warnings when you compile your L^AT_EX code. These are not as serious as errors, but you should still fix them if possible. Again, your editor will give you a list of these warnings.

4.3 Bad Boxes

If you have Bad Boxes when you compile your code, that means that somewhere in your document, the content does not fit correctly into the space where it is supposed to be. You'll have a list of these as well.

5 Example Document

Here's an example of the code for a complete document. You can copy this into a new document in your editor and try to compile it to see if you have things set up correctly.

```
\documentclass[11pt]{article}
\usepackage{amsmath, amssymb, amscd, amsthm, mathrsfs, url}
\usepackage{fancyhdr}
\usepackage[left=.5in,top=.75in,right=.5in,bottom=.75in]{geometry}

\pagestyle{fancy}
\lhead{Getting Started with \LaTeX{}}
\rhead{Rebekah Yates}

\begin{document}

Euler's equation
\begin{equation*}
e^{i\pi} = -1.
\end{equation*}
is considered by many to be one of the most beautiful equations in mathematics
because it relates several fundamental constants and is very simple.

\end{document}
```

6 Environments

One very useful concept in using L^AT_EX is the idea of an environment. There are many different environments you may find helpful; several examples are below.

6.1 Itemize

If you want a bulleted list, use the `itemize` environment. Here's an example:

```
\begin{itemize}
\item This is the first item in the bulleted list.
\item This is the second item in the bulleted list.
\item This is the third item in the bulleted list.
\end{itemize}
```

When compiled, this code appears as

- This is the first item in the bulleted list.
- This is the second item in the bulleted list.
- This is the third item in the bulleted list.

You can also nest `itemize` environments:

```
\begin{itemize}
\item This is the first item in the bulleted list.
\begin{itemize}
\item This is a subitem.
\item This is another subitem.
\end{itemize}
\item This is the second item in the bulleted list.
\item This is the third item in the bulleted list.
\end{itemize}
```

When compiled, this code looks like

- This is the first item in the bulleted list.
 - This is a subitem.
 - This is another subitem.
- This is the second item in the bulleted list.
- This is the third item in the bulleted list.

6.2 Enumerate

If you would prefer a numbered list, you can use an `enumerate` environment:

```

\begin{enumerate}
\item This is the first item in the list.
\begin{itemize}
\item This is a subitem.
\item This is another subitem.
\end{itemize}
\item This is the second item in the list.
\item This is the third item in the list.
\end{enumerate}

```

The output for this code is

1. This is the first item in the list.
 - (a) This is a subitem.
 - (b) This is another subitem.
2. This is the second item in the list.
3. This is the third item in the list.

Note that L^AT_EX automatically chooses how to enumerate your items for you. If you want to choose the characters yourself, you can use an optional command (in square brackets) for each item. For example,

```

\begin{enumerate}
\item[(I)] Sometimes Roman numerals are nice.
\item[(A)] Sometimes I prefer capital letters.
\item[1.2.3.] Often I just want to use the number of the problem for which I'm writing a solution.
\end{enumerate}

```

will compile as

- (I) Sometimes Roman numerals are nice.
- (A) Sometimes I prefer capital letters.
- 1.2.3. Often I just want to use the number of the problem for which I'm writing a solution.

6.3 Theorems and Proofs

You can set theorems, corollaries, propositions, etc., apart nicely if you include the following new command definitions in your preamble:

```

\newtheorem{thm}{Theorem}
\newtheorem{defn}{Definition}
\newtheorem{cor}[thm]{Corollary}
\newtheorem{lemma}[thm]{Lemma}
\newtheorem{prop}[thm]{Proposition}

```

After a theorem, you can put its proof in a proof environment, too. Here's an example:

```
\begin{thm}
Let  $F(x)$  be an antiderivative of  $f(x)$  on  $(a, b)$ .
Then every other antiderivative on  $(a, b)$  is of the form  $F(x) + C$  for some
constant  $C$ .
\end{thm}

\begin{proof}
Let  $F(x)$  and  $G(x)$  be antiderivatives of  $f(x)$  and set  $H(x) = G(x) - F(x)$ .
Then  $H'(x) = G'(x) - F'(x) = f(x) - f(x) = 0$ . By a corollary to the Mean Value
Theorem,  $H(x)$  must be constant; i.e., there is a  $C \in \mathbb{R}$  such that
 $H(x) = C$ , so  $G(x) = F(x) + C$ .
\end{proof}
```

Compiled, this looks like

Theorem 1. *Let $F(x)$ be an antiderivative of $f(x)$ on (a, b) . Then every other antiderivative on (a, b) is of the form $F(x) + C$ for some constant C .*

Proof. Let $F(x)$ and $G(x)$ be antiderivatives of $f(x)$ and set $H(x) = G(x) - F(x)$. Then $H'(x) = G'(x) - F'(x) = f(x) - f(x) = 0$. By a corollary to the Mean Value Theorem, $H(x)$ must be constant; i.e., there is a $C \in \mathbb{R}$ such that $H(x) = C$, so $G(x) = F(x) + C$. \square

7 Other Resources

This document is only intended to give you a start on learning L^AT_EX. Of course, the best way to learn L^AT_EX is by using L^AT_EX, but there are many other, far better and more comprehensive resources available to you. Here are some that I (or others) have found useful:

- *The Not So Short Guide to L^AT_EX 2 ϵ* , at <http://tobi.oetiker.ch/lshort/lshort.pdf>, is an extremely useful resource for most things L^AT_EX.
- *The Comprehensive List of L^AT_EX Symbols* at <http://www.ctan.org/tex-archive/info/symbols/comprehensive/symbols-a4.pdf> is just what it says it is.
- DeT_EXify, at <http://detexify.kirelabs.org/classify.html>, allows you to draw a symbol for which you do not know the L^AT_EX code and generates suggestions for you.
- T_EX Users Group at <http://tug.org/> has some helpful resources and examples for T_EX users new and experienced.
- The Comprehensive T_EX Archive Network at <http://www.ctan.org/> has all kinds of things.
- The American Mathematical Society has a short guide to using L^AT_EX at <http://www.ams.org/tex/short-math-guide.html>.
- The T_EX Stack Exchange at <http://tex.stackexchange.com/> is a place where you can ask questions and see questions and answers asked by other users of T_EX and L^AT_EX. If you want to answer questions, you need to be a member (and you're free to become one), but anyone can ask questions.