

The L^AT_EX document preparation system

Vineet Padmanabhan

School of Computer & Information Sciences
University of Hyderabad

Table of Contents

- 1 Introduction
- 2 Why Even Bother...
- 3 Show
- 4 A Basic Document
 - Writing L^AT_EXcode
 - Basic Formatting
- 5 L^AT_EX and You
 - The Files
- 6 L^AT_EX Extended
 - Common Errors
 - Macros
- 7 Conclusion

Table of Contents

- 1 Introduction
- 2 Why Even Bother...
- 3 Show
- 4 A Basic Document
 - Writing L^AT_EXcode
 - Basic Formatting
- 5 L^AT_EX and You
 - The Files
- 6 L^AT_EX Extended
 - Common Errors
 - Macros
- 7 Conclusion

Name of the game

T_EX

typesetting system developed by Donald E. Knuth (Stanford University) to create beautiful documents, especially those containing maths. T_EX is free software with copyright vested in the American Mathematical Society.

L^AT_EX

T_EX-macroprocessor written by Leslie Lamport, which implements a markup-language (similar: HTML, XML). Users can concentrate on the structure of their document rather than on formatting.

L^AT_EX is...

... a sophisticated document preparation system.

L^AT_EX has...

- Stylistic uniformity
- Bibliography support
- Sophisticated structuring abilities
- Reference tracking
- Highly extendible capabilities

L^AT_EX is not...

... a word processor.

L^AT_EX does not...

- Spell-check your documents^a
- Give you complete control over formatting
- Provide a graphical interface for editing

^aYou can use ispell to check your L^AT_EX

"You take care of writing, and we'll take care of presentation."

Why L^AT_EX?

Presentation shouldn't get in the way of content.

For example...

- With a word processor, you spend valuable time agonizing over what font size to make the section headings.
With L^AT_EX, you just tell it to start a new section.
- With a word processor, changing the formatting means you have to change each instance individually.
With L^AT_EX, you just redefine the relevant commands.
- With a word processor, you have to carefully match any provided templates.
With L^AT_EX, you can be sure you've fit the template, and switch templates easily.

Table of Contents

- 1 Introduction
- 2 Why Even Bother...
- 3 Show
- 4 A Basic Document
 - Writing L^AT_EXcode
 - Basic Formatting
- 5 L^AT_EX and You
 - The Files
- 6 L^AT_EX Extended
 - Common Errors
 - Macros
- 7 Conclusion

It Generates Beautiful Equations from Ugly Text...

```
\int^{+\infty}_{-\infty} x^{2n} e^{-\alpha x^2} dx  
= \sqrt{\frac{\pi}{\alpha}}  
\alpha^{-n} \frac{2n!}{2^{2n} n!}
```

$$\int_{-\infty}^{+\infty} x^{2n} e^{-\alpha x^2} dx = \sqrt{\frac{\pi}{\alpha}} \alpha^{-n} \frac{2n!}{2^{2n} n!}$$

...It can also generate cute figures from ugly text.

Linux People Say...



Image ./Docs/WindowsLogo (Color JPG)

Linux People Also Say...



Courtesy : ISmailo Dabo

Have You Ever Seen a Frozen Penguin?



Courtesy : ISmailo Dabo



MS Word

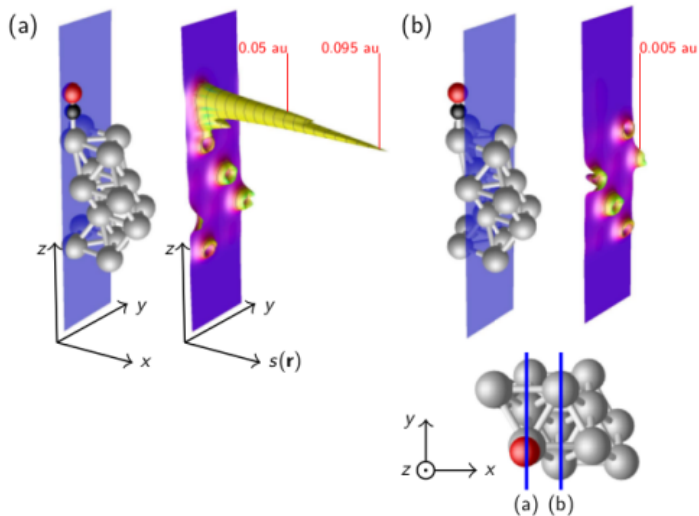
- Is Intuitive...
- You see what you do...
- Is user-friendly (no need to *patiently* compile *twice*)...
- Is *a priori* quicker...
- ...

LaTeX

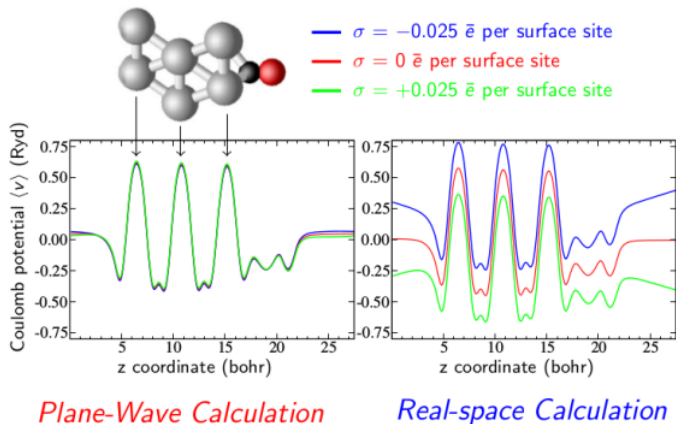
- Does not take initiatives...
- Is free...
- Is suited for scientific editing...
- Is scriptizable...
- ...

Table of Contents

- 1 Introduction
- 2 Why Even Bother...
- 3 Show**
- 4 A Basic Document
 - Writing L^AT_EXcode
 - Basic Formatting
- 5 L^AT_EX and You
 - The Files
- 6 L^AT_EX Extended
 - Common Errors
 - Macros
- 7 Conclusion

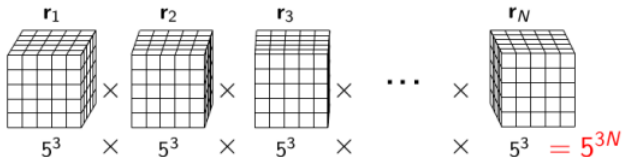


Courtesy : ISmailo Dabo



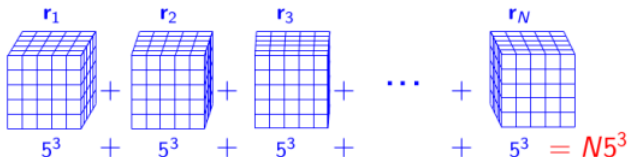
Courtesy : ISmailo Dabo

$$\hat{H}\Psi(\mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_3, \dots, \mathbf{r}_N) = E\Psi(\mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_3, \dots, \mathbf{r}_N)$$



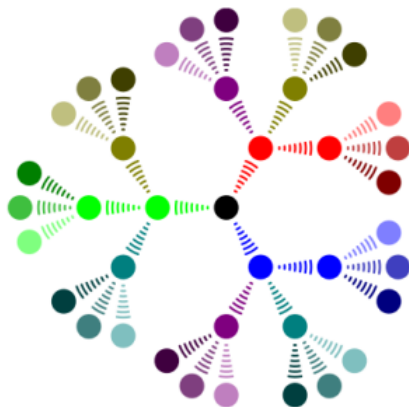
$$5^3 \times 5^3 \times 5^3 \times \dots \times 5^3 = 5^{3N}$$

$$\hat{h}^{eff} \psi_1(\mathbf{r}_1) = \epsilon_1 \psi(\mathbf{r}_1), \dots, \hat{h}^{eff} \psi_N(\mathbf{r}_N) = \epsilon_N \psi(\mathbf{r}_N)$$

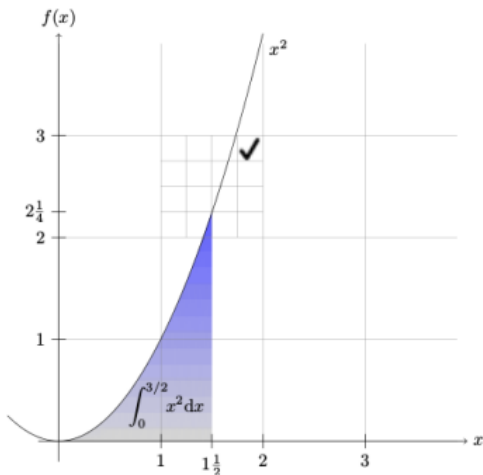


$$5^3 + 5^3 + 5^3 + \dots + 5^3 = N5^3$$

Courtesy : ISmailo Dabo



Courtesy : ISmailo Dabo



Courtsey : ISmailo Dabo

Table of Contents

- 1 Introduction
- 2 Why Even Bother...
- 3 Show
- 4 A Basic Document**
 - Writing L^AT_EXcode
 - Basic Formatting
- 5 L^AT_EXand You
 - The Files
- 6 L^AT_EXExtended
 - Common Errors
 - Macros
- 7 Conclusion

“Hello L^AT_EX!”

Creating a L^AT_EX Document

- Write a .tex file using any text editor and save it in a relevant folder

```
% this is hello.tex
\documentclass{article}
\begin{document}
  Hello, \LaTeX!
\end{document}
```

- Compile using L^AT_EX.

```
$ cd ~/Desktop/Latex-IEEE/
$ latex hello.tex
$ dvips -P pdf hello.dvi
$ ps2pdf hello.ps
$ pdflatex hello.tex
```

- Preview the results

```
$ evince hello.pdf &
```

documentclass

L^AT_EX has several templates, selected using

```
\documentclass
```

Classes

- book
- report
- article
- letter
- beamer

Etc.

You'll be using the **article** class for your paper, **beamer** class for your presentation

Declarations and Environments

Declarations...

- Are stated once
- Take effect until further notice
- Can optionally be constrained

Ex. `\documentclass`, `\small`

Environments...

- Have matching begin and end declarations
- Must be constrained

Ex. `\begin{document}` ... `\end{document}`

Arguments

Required arguments...

- Are contained in curly braces
- Must be included

Ex. `\documentclass{article}`

Optional arguments...

- Are contained in square brackets
- Can be left out
- Give you more control over the commands

Ex. `\documentclass[12pt]{article}`

Special Characters

- Another type of command
- Don't define any formatting or structure
- Print non-standard characters or characters which usually mean something else

Ex. `\LaTeX`, `\textbackslash`, `\%`

Note: `%` is a special character reserved for comments (after a `%`, the rest of a line is ignored by the compiler)

Packages

Packages allow you to further customize L^AT_EX

The command:

```
\usepackage{name}
```

Some packages:

graphicx, epsfig, geometry, fancyhdr, setspace, amsmath, listings, xcolor, url...

Most of the packages you'll need are already included in the template

Font Types

Font face:

```
\emph{Text}, \textbf{Text}, \texttt{Text}, \textrm{Text},  
\textsf{Text}, \textsc{Text}
```

Text, **Text**, Text, Text, Text, TEXT

Font size:

```
{\tiny Text}, {\scriptsize Text}, {\footnotesize Text},  
{\small Text}, {\normalsize Text}, {\large Text }, {\Large  
Text}, {\LARGE Text}, {\huge Text}, {\Huge Text}
```

Text, Text, Text, Text, Text, Text , Text, Text, Text, Text

Alignment

Alignment:

```
\begin{center}  
center  
\end{center}  
\begin{flushleft}  
flushleft  
\end{flushleft}  
\begin{flushright}  
flushright  
\end{flushright}
```

center

flushleft

flushright

Bold text, Underlined, Emphasized

Some of the `\textbf{greatest}`
discoveries in `\underline{science}`
were made by `\textbf{\textit{accident}}`.

Some of the **greatest** discoveries in science were
made by *accident*.

Bold text, underlined, Emphasized

Some of the greatest `\emph{discoveries}`
in science were made by accident.

`\textit{Some of the greatest \emph{discoveries}`
in science were made by accident.}

`\textbf{Some of the greatest \emph{discoveries}`
in science were made by accident.}

Some of the greatest *discoveries* in science were
made by accident.

*Some of the greatest discoveries in science were
made by accident.*

**Some of the greatest *discoveries* in science
were made by accident.**

Setting Fonts

```
\documentclass[12pt]{article}  
\usepackage{fontspec}
```

```
\setmainfont{Times New Roman}
```

```
\title{Sample font document}  
\author{Hubert Farnsworth}  
\date{this month, 2014}
```

```
\begin{document}
```

```
\maketitle
```

This an `\textit{example}` of document compiled
with `\textbf{xelatex}` compiler. LuaLaTeX should
work fine also.

```
\end{document}
```

Sample font document

Hubert Farnsworth

this month, 2014

This an *example* of document compiled with **xelatex** compiler. LuaLaTeX should work fine also.

Setting fonts for different latex elements

```
\documentclass[12pt]{article}
\usepackage{fontspec}

%This would work on a standard latex installation, \\
%(your local computer)
%-----
\setromanfont{Times New Roman}
\setsansfont{Arial}
\setmonofont[Color={0019D4}]{Courier New}
%-----

\title{Sample font document}
\author{Hubert Farnsworth}
\date{this month, 2014}
```

Setting fonts for different latex elements

```
\begin{document}
```

```
\maketitle
```

This an `\textit{example}` of document compiled with
`\textbf{xelatex}` compiler. If you have to write some code

```
\begin{verbatim}
```

usually this environment is used to display code

```
<html>
```

```
<head> </head>
```

```
<body>
```

```
<h1> Hello World</h1>
```

```
</body>
```

```
</html>
```

```
{\sffamily This is a sample text in \textbf{Sans Serif Font Typ
```

```
\end{document}
```

Sample font ocument

Hubert Farnsworth

this month, 2014

This an *example* of document compiled with **xelatex** compiler. If you have to write some code

usually this environment is used to display code

```
<html>
<head> </head>
<body>
<h1> Hello World</h1>
</body>
</html>
```

This is a sample text in **Sans Serif Font Typeface**

Spacing

Margins

The default: between 1.5 inches and 1.875 inches Setting margins:

```
\usepackage[margin=0.5in]{geometry}
```

Paragraphs and other breaks

Paragraphs are separated by a blank line.

You can force a new line using `\\`

To force a new page, use. `\newpage` or `\clearpage`

Other spacing

Force a space using `~`

Add space using `\hspace{1in}` or `\vspace{1in}`

Fill space using `\hfill` or `\vfill`

Paragraphs and New Lines

```
\begin{center}
```

Example 1: The following paragraph (given in quotes) is an example of Center Alignment using the center environment.

```
“LaTeX is a document preparation system and document markup  
language. LaTeX uses the TeX typesetting program for formatting  
its output, and is itself written in the TeX macro language.  
LaTeX is not the name of a particular editing program, but  
refers to the encoding or tagging conventions that are used  
in LaTeX documents”.
```

```
\end{center}
```

Example 1: The following paragraph (given in quotes) is an example of Center Alignment using the center environment.

“LaTeX is a document preparation system and document markup language. LaTeX uses the TeX typesetting program for formatting its output, and is itself written in the TeX macro language. LaTeX is not the name of a particular editing program, but refers to the encoding or tagging conventions that are used in LaTeX documents”.

Paragraph Alignment

```
\begin{flushleft}
```

```
“LaTeX is a document preparation system and document markup  
language. LaTeX uses the TeX typesetting program for formatting  
its output, and is itself written in the TeX macro language.  
LaTeX is not the name of a particular editing program, but refer  
to the encoding or tagging conventions that are used in LaTeX do  
\end{flushleft}
```

“LaTeX is a document preparation system and document markup language. LaTeX uses the TeX typesetting program for formatting its output, and is itself written in the TeX macro language. LaTeX is not the name of a particular editing program, but refers to the encoding or tagging conventions that are used in LaTeX documents”.

Paragraph Indentation

```
\setlength{\parindent}{10ex}
```

This is the text in first paragraph. This is the text in first paragraph. This is the text in first paragraph. \par

```
\noindent %The next paragraph is not indented
```

This is the text in second paragraph. This is the text in second paragraph. This is the text in second paragraph.

This is the text in first paragraph. This is the text in first paragraph.

This is the text in first paragraph.

This is the text in second paragraph. This is the text in second paragraph. This is the text in second paragraph.

Lengths in Latex

```
\documentclass{article}
\usepackage[utf8]{inputenc}
\usepackage{amssymb}
\usepackage{graphicx}

\begin{document}
Example of a picture with different lengths
\includegraphics[width=15ex]{lion-logo}

\includegraphics[width=15em]{lion-logo}
\end{document}
```

ShareLaTeX



Learn

ShareLaTeX



Learn

Setting Lengths

```
\setlength{\lengthname}{value_in_specified_unit}
```

```
\setlength{\columnsep}{1in}
```

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut portitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit

fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus

Headers & Footers

```
\documentclass[a4paper,12pt,twoside]{book}
\usepackage[english]{babel}
\usepackage[utf8]{inputenc}
```

```
\pagestyle{headings}
```

```
\begin{document}
\chapter{Sample Chapter}
\section{New section}
```

Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like “Huardest gefburn? Kjift ” not at all!...

```
\end{document}
```

Chapter 2

Sample Chapter

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

2.1 New section

Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like "Haudet gellum"? Kjff - not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

This is the second paragraph. Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like "Haudet gellum"? Kjff - not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

And after the second paragraph follows the third paragraph. Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like "Haudet gellum"? Kjff - not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

After this fourth paragraph, we start a new paragraph sequence. Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like "Haudet gellum"? Kjff - not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between

Setting Page Style

```
\documentclass{article}
\usepackage[english]{babel}
\usepackage[utf8]{inputenc}
\usepackage{fancyhdr}
\pagestyle{fancy}
\fancyhf{}
\rhead{Overleaf}
\lhead{Guides and tutorials}
\rfoot{Page \thepage}
```

```
\begin{document}
\section{First Section}
```

Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this ..overleaf-header-footer

```
\end{document}
```

1 First Section

Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like “Huaardeest gefburn”? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

Style Customisation

```
\documentclass[a4paper,12pt,twoside]{book}
\usepackage[utf8]{inputenc}
\usepackage[english]{babel}
\usepackage{fancyhdr}

\pagestyle{fancy}
\fancyhf{}
\fancyhead[LE,R0]{Overleaf}
\fancyhead[RE,L0]{Guides and tutorials}
\fancyfoot[CE,C0]{\leftmark}
\fancyfoot[LE,R0]{\thepage}

\begin{document}

\chapter{Using different page styles}
Lorem ipsum dolor sit amet, consectetur adipiscing ...
```

2.1 New section

Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like “Huardest gefburn”? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

This is the second paragraph. Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like “Huardest gefburn”? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

And after the second paragraph follows the third paragraph. Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like “Huardest gefburn”? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

After this fourth paragraph, we start a new paragraph sequence. Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like “Huardest gefburn”? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between

this text and some nonsense like “Huardest gefburn”? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

Page Numbering

```
\documentclass{article}
\usepackage[utf8]{inputenc}

\pagenumbering{roman}

\begin{document}
\tableofcontents

\section{Testing section}
...

\end{document}
```


Contents

1	Testing section	i
2	Heading on level 1 (section)	ii
2.1	Heading on level 2 (subsection)	iii
2.1.1	Heading on level 3 (subsection)	v
3	Lists	vii
3.1	Example for list (itemize)	vii
3.1.1	Example for list (4*itemize)	ix
3.2	Example for list (enumerate)	xi
3.2.1	Example for list (4*enumerate)	xii
3.3	Example for list (description)	xiv
3.3.1	Example for list (4*description)	xvi

Numbering Styles

```
\usepackage[utf8]{inputenc}
```

```
\pagenumbering{alph}
```

```
\begin{document}
```

```
\tableofcontents
```

```
\section{Testing section}
```

```
...
```

```
\end{document}
```

Contents

1	Testing section	a
2	Heading on level 1 (section)	b
2.1	Heading on level 2 (subsection)	c
2.1.1	Heading on level 3 (subsection)	e
3	Lists	g
3.1	Example for list (itemize)	g
3.1.1	Example for list (4*itemize)	i
3.2	Example for list (enumerate)	k
3.2.1	Example for list (4*enumerate)	l
3.3	Example for list (description)	n
3.3.1	Example for list (4*description)	p

Lists

There are two main types...

Bulleted lists:

```
\begin{itemize}  
\item Text  
\item Text  
\end{itemize}
```

- Text
- Text

Numbered lists:

```
\begin{enumerate}  
\item Text \hspace{.2in}  
\item Text  
\end{enumerate}
```

- 1 Text
- 2 Text

Using Listings to highlight code

```
\begin{lstlisting}
import numpy as np

def incmatrix(genl1,genl2):
    m = len(genl1)
    n = len(genl2)
    M = None #to become the incidence matrix
    VT = np.zeros((n*m,1), int) #dummy variable

    #compute the bitwise xor matrix
    M1 = bitxormatrix(genl1)
    M2 = np.triu(bitxormatrix(genl2),1)

    for i in range(m-1):
        for j in range(i+1, m):
            [r,c] = np.where(M2 == M1[i,j])
            for k in range(len(r)):
                VT[(i)*n + r[k]] = 1;
                VT[(i)*n + c[k]] = 1;
                VT[(j)*n + r[k]] = 1;
                VT[(j)*n + c[k]] = 1;

            if M is None:
                M = np.copy(VT)
            else:
                M = np.concatenate((M, VT), 1)

            VT = np.zeros((n*m,1), int)

    return M
\end{lstlisting}
```

```

import numpy as np

def incmatrix(genl1, genl2):
    m = len(genl1)
    n = len(genl2)
    M = None #to become the incidence matrix
    VT = np.zeros((n*m,1), int) #dummy variable

    #compute the bitwise xor matrix
    M1 = bitxormatrix(genl1)
    M2 = np.triu(bitxormatrix(genl2), 1)

    for i in range(m-1):
        for j in range(i+1, m):
            [r, c] = np.where(M2 == M1[i, j])
            for k in range(len(r)):
                VT[(i)*n + r[k]] = 1;
                VT[(i)*n + c[k]] = 1;
                VT[(j)*n + r[k]] = 1;
                VT[(j)*n + c[k]] = 1;

            if M is None:
                M = np.copy(VT)
            else:
                M = np.concatenate((M, VT), 1)

```

Importing code from a file

The next code will be directly imported from a file

```
\lstinputlisting[language=Octave]{BitXorMatrix.m}
```

The next code will be directly imported from a file:

```
function X = BitXorMatrix(A,B)
%function to compute the sum without charge of two vectors

%convert elements into unsigned integers
A = uint8(A);
B = uint8(B);

m1 = length(A);
m2 = length(B);
X = uint8(zeros(m1, m2));
for n1=1:m1
    for n2=1:m2
        X(n1, n2) = bitxor(A(n1), B(n2));
    end
end
```

The command `\lstinputlisting[language=Octave]{BitXorMatrix.m}` `\%` imports the code from the file `BitXorMatrix.m`, `\%` the additional parameter in between brackets `\%` enables language highlighting for the Octave programming language. If you need to import only part of the `\%` file you can specify two comma-separated `\%` parameters inside the brackets. `\%` For instance, to import the code from `\%` the line 2 to the line 12, the previous command becomes `\%`

```
\lstinputlisting[language=Octave, firstline=2, lastline=12]{BitX
```

If `firstline` or `lastline` is omitted, it's assumed `\%` that the values are the beginning of the file, `\%` or the bottom of the file, respectively.

Code Styles & Colours

```
\documentclass{article}
\usepackage[utf8]{inputenc}

\usepackage{listings}
\usepackage{xcolor}

\definecolor{codegreen}{rgb}{0,0.6,0}
\definecolor{codegray}{rgb}{0.5,0.5,0.5}
\definecolor{codepurple}{rgb}{0.58,0,0.82}
\definecolor{backcolour}{rgb}{0.95,0.95,0.92}

\lstdefinestyle{mystyle}{
  backgroundcolor=\color{backcolour},
  commentstyle=\color{codegreen},
  keywordstyle=\color{magenta},
  numberstyle=\tiny\color{codegray},
  stringstyle=\color{codepurple},
  basicstyle=\ttfamily\footnotesize,
  breakatwhitespace=false,
  breaklines=true,
  captionpos=b,
  keepspaces=true,
  numbers=left,
  numbersep=5pt,
  showspaces=false,
  showstringspaces=false,
  showtabs=false,
  tabsize=2
}
\lstset{style=mystyle}

\begin{document}
The next code will be directly imported from a file

\lstinputlisting[language=Octave]{BitXorMatrix.m}
\end{document}
```

The next code will be directly imported from a file:

```
1 function X = BitXorMatrix(A,B)
2 %function to compute the sum without charge of two vectors
3
4 %convert elements into unsigned integers
5 A = uint8(A);
6 B = uint8(B);
7
8 m1 = length(A);
9 m2 = length(B);
10 X = uint8(zeros(m1, m2));
11 for n1=1:m1
12     for n2=1:m2
13         X(n1, n2) = bitxor(A(n1), B(n2));
14     end
15 end
```

Code highlighting with minted

```

\documentclass{article}
\usepackage[utf8]{inputenc}
\usepackage[english]{babel}

\usepackage{minted}

\begin{document}
\begin{minted}{python}
import numpy as np

def incmatrix(genl1,genl2):
    m = len(genl1)
    n = len(genl2)
    M = None #to become the incidence matrix
    VT = np.zeros((n*m,1), int) #dummy variable

    #compute the bitwise xor matrix
    M1 = bitxormatrix(genl1)
    M2 = np.triu(bitxormatrix(genl2),1)

    for i in range(m-1):
        for j in range(i+1, m):
            [r,c] = np.where(M2 == M1[i,j])
            for k in range(len(r)):
                VT[(i)*n + r[k]] = 1;
                VT[(i)*n + c[k]] = 1;
                VT[(j)*n + r[k]] = 1;
                VT[(j)*n + c[k]] = 1;

            if M is None:
                M = np.copy(VT)
            else:
                M = np.concatenate((M, VT), 1)

    VT = np.zeros((n*m,1), int)

return M

```

```
import numpy as np

def incmatrix(genl1,genl2):
    m = len(genl1)
    n = len(genl2)
    M = None #to become the incidence matrix
    VT = np.zeros((n*m,1), int) #dummy variable

    #compute the bitwise xor matrix
    M1 = bitxormatrix(genl1)
    M2 = np.triu(bitxormatrix(genl2),1)

    for i in range(m-1):
        for j in range(i+1, m):
            [r,c] = np.where(M2 == M1[i,j])
            for k in range(len(r)):
                VT[(i)*n + r[k]] = 1;
                VT[(i)*n + c[k]] = 1;
                VT[(j)*n + r[k]] = 1;
                VT[(j)*n + c[k]] = 1;

            if M is None:
                M = np.copy(VT)
            else:
                M = np.concatenate((M, VT), 1)

    VT = np.zeros((n*m,1), int)
```

Basic Usage

```

\begin{minted}
[
frame=lines,
framesep=2mm,
baselinestretch=1.2,
bgcolor=LightGray,
fontsize=\footnotesize,
linenos
]
{python}
import numpy as np

def incmatrix(genl1,genl2):
    m = len(genl1)
    n = len(genl2)
    M = None #to become the incidence matrix
    VT = np.zeros((n*m,1), int) #dummy variable

    #compute the bitwise xor matrix
    M1 = bitxormatrix(genl1)
    M2 = np.triu(bitxormatrix(genl2),1)

    for i in range(m-1):
        for j in range(i+1, m):
            [r,c] = np.where(M2 == M1[i,j])
            for k in range(len(r)):
                VT[(i)*n + r[k]] = 1;
                VT[(i)*n + c[k]] = 1;
                VT[(j)*n + r[k]] = 1;
                VT[(j)*n + c[k]] = 1;

            if M is None:
                M = np.copy(VT)
            else:
                M = np.concatenate((M, VT), 1)

    VT = np.zeros((n*m,1), int)

```

```

1  import numpy as np
2
3  def incmatrix(genl1,genl2):
4      m = len(genl1)
5      n = len(genl2)
6      M = None #to become the incidence matrix
7      VT = np.zeros((n*m,1), int) #dummy variable
8
9      #compute the bitwise xor matrix
10     M1 = bitxormatrix(genl1)
11     M2 = np.triu(bitxormatrix(genl2),1)
12
13     for i in range(m-1):
14         for j in range(i+1, m):
15             [r,c] = np.where(M2 == M1[i,j])
16             for k in range(len(r)):
17                 VT[(i)*n + r[k]] = 1;
18                 VT[(i)*n + c[k]] = 1;
19                 VT[(j)*n + r[k]] = 1;
20                 VT[(j)*n + c[k]] = 1;
21
22     if M is None:
23         M = np.copy(VT)
24     else:
25         M = np.concatenate((M, VT), 1)
26
27     VT = np.zeros((n*m,1), int)
28
29     return M

```

Hyperlinks

```
\documentclass{book}
\usepackage[utf8]{inputenc}
\usepackage[english]{babel}
```

```
\usepackage{hyperref}
\hypersetup{
  colorlinks=true,
  linkcolor=blue,
  filecolor=magenta,
  urlcolor=cyan,
}
```

```
\urlstyle{same}
```

```
\begin{document}
```

```
\tableofcontents
```

```
\chapter{First Chapter}
```

This will be an empty chapter and I will put some text here

```
\begin{equation}
\label{eq:1}
\sum_{i=0}^{\infty} a_i x^i
\end{equation}
```

The equation `\ref{eq:1}` shows a sum that is divergent. This formula will later be used in the page `\pageref{second}`.

For further references see `\href{http://www.sharelatex.com}{Something Linky}` or go to the next url: `\url{http://www.sharelatex.com}` or open the next file `\href{run:./file.txt}{File.txt}`

It's also possible to link directly any word or `\hyperlink{thesentence}{any sentence}` in your document.

First Chapter

This will be an empty chapter and I will put some text here

$$\sum_{i=0}^{\infty} a_i x^i \tag{1.1}$$

The equation 1.1 shows a sum that is divergent. This formula will be latter used in the page 5.

For further references see [Something Linky](#) or go to the next url: <http://www.sharelatex.com> or open the next file [File.txt](#)

It's also possible to link directly any word or [any sentence](#) in your document.

Linking web addresses

For further references see `\href{http://www.sharelatex.com}{Some`
or go to the next url: `\url{http://www.sharelatex.com}`

For further references see **Something Linky** or go to the next url:
<http://www.sharelatex.com>

Linking local files

For further references see `\href{http://www.sharelatex.com}{Something Linky}` or go to the next url: `\url{http://www.sharelatex.com}` or open the next file `\href{run:./file.txt}{File.txt}`

For further references see **Something Linky** or go to the next url:
<http://www.sharelatex.com> or open the next file **[File.txt](#)**

Footnotes, endnotes, marginal notes

```
\footnote{footnote text}
```

```
\endnote{endnote text}
```

```
\marginpar{marginal note}
```

Footnote

I'm writing something here to test `\footnote[10]{footnotes working fine}` several features. You can write the footnote text`\footnotemark` in its own line.
`\footnotetext{Second footnote}`

I'm writing something here to test ¹⁰ several features. You can write the footnote text¹ in its own line.

¹⁰footnotes working fine

¹Second footnote

Margin Notes

`\marginpar{The new section start point will be pointed out by an arrow}`

at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

1 Introduction

Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like “Huardest gefburn”? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

This is the second paragraph. Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like “Huardest gefburn”? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

And after the second paragraph follows the third paragraph. Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this

The new section start point will be pointed out by an arrow

Margin Note package

```
\documentclass{article}  
\usepackage[utf8]{inputenc}  
\usepackage[english]{babel}
```

```
\usepackage{geometry}  
\usepackage{marginnote}
```

```
\begin{document}  
...
```

```
\marginnote{This is a margin note using the geometry package,  
set at 3cm vertical offset to the line it is typesetted.}[3cm]  
...
```

written in of the original language. There is no need for special content, but the length of words should match the language.

1 Introduction

Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like “Huardest gefburn”? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

This is the second paragraph. Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like “Huardest gefburn”? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

And after the second paragraph follows the third paragraph. Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like “Huardest gefburn”? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

This is a margin note using the geometry package, set at 5cm vertical offset to the first line it is typeset.

Using Colours in Latex

```
\documentclass{article}
\usepackage[utf8]{inputenc}
\usepackage[english]{babel}

\usepackage{xcolor}

\begin{document}
```

This example shows different examples on how to use the `\texttt{xcolor}` package to change the colour of elements in `\LaTeX`.

```
\begin{itemize}
\color{blue}
\item First item
\item Second item
\end{itemize}

\noindent
{\color{red} \rule{\linewidth}{0.5mm} }

\end{document}
```


This example shows different examples on how to use the `xcolor` package to change the colour of elements in L^AT_EX.

- First item
- Second item



Basic Usage

```
\documentclass{article}
\usepackage[utf8]{inputenc}
\usepackage[english]{babel}

\usepackage[dvipsnames]{xcolor}

\begin{document}
```

This example shows different examples on how to use the `\texttt{xcolor}` package to change the colour of elements in L^AT_EX.

```
\begin{itemize}
\color{ForestGreen}
\item First item
\item Second item
\end{itemize}

\noindent
{\color{RubineRed} \rule{\linewidth}{0.5mm} }
```

The background colour of some text can also be `\textcolor{red}{easily}` set. For instance, you can change to orange the background of `\colorbox{BurntOrange}{this text}` and then continue typing.

```
\end{document}
```

This example shows different examples on how to use the `xcolor` package to change the colour of elements in L^AT_EX.

- First item
- Second item

The background colour of some text can also be easily set. For instance, you can change to orange the background of `this text` and then continue typing.

Creating your own colours

```
\documentclass{article}
\usepackage[utf8]{inputenc}
\usepackage[english]{babel}

\usepackage[dvipsnames]{xcolor}

\definecolor{mypink1}{rgb}{0.858, 0.188, 0.478}
\definecolor{mypink2}{RGB}{219, 48, 122}
\definecolor{mypink3}{cmyk}{0, 0.7808, 0.4429, 0.1412}
\definecolor{mygray}{gray}{0.6}

\begin{document}
User-defined colours with different colour models:

\begin{enumerate}
\item \textcolor{mypink1}{Pink with rgb}
\item \textcolor{mypink2}{Pink with RGB}
\item \textcolor{mypink3}{Pink with cmyk}
\item \textcolor{mygray}{Gray with gray}
\end{enumerate}

\end{document}
```

User-defined colours with different colour models:

1. Pink with `rgb`
2. Pink with `RGB`
3. Pink with `cmyk`
4. Gray with `gray`

- `rgb` Three comma-separated values between 0 and 1
- `RGB` the numbers are integers between 0 and 255.
- `cmyk` Cyan, Magenta, Yellow and black between 0 and 1
- `gray` Grey scale. A single number between 0 and 1.

Setting the page background colour

```
\pagecolor{black}  
\color{white}
```

This document present several examples on how to use the `color` package to change the colour of elements in L^AT_EX.

- First item
- Second item

Not only blocks, such as environments, can be set to a determined colour, but some **special words** too. You can even use your own user-defined colours. Below the same colour with different models:

1. Pink with `rgb`
2. Pink with `RGB`
3. Pink with `cmyk`
4. Gray with `gray`

The background colour of some text can also be **easily** set. For instance, you can change to orange the background of `this text` and then continue typing.

Document Classes and Document Structure

You mark out the structural elements, L^AT_EX sets them:

First, the *preamble*

```
\documentclass[12pt]{article}
\usepackage{chicago}
\usepackage{times,geometry,makeidx,multicol}
\geometry{left=1in,right=1in,
top=1in,bottom=1in}
\title{Lincoln's Peoria Speech of 1854}
\author{John Burt}\date{\today}
\pagestyle{myheadings}
\markboth{Lincoln's Peoria Speech}
{Lincoln's Peoria Speech}
\makeindex
```

Document Classes and Document Structure

Then the body

```
\begin{document}  
\maketitle  
\tableofcontents  
\section{The problem of  
extreme moral conflict}  
blah\index{Federalist@\emph{The Federalist}}.
```

Blank lines start new paragraphs.

```
blah\cite{Douglas1854}  
\section{The irony of American History}  
blah blah blah\footnote{footnote text}
```


Document Classes and Document Structure

Then the ending

```
\bibliographystyle{chicago}  
\bibliography{lincoln}  
\printindex  
\end{document}
```

Bibliography

biblio.bib acts as a database of references, and only includes in the bibliography those references you cite in your paper

BibT_EX

```
@article{nameofentry,  
author = {John Backus},  
title = {Symmetric Encryption},  
journal = {Journal of Modalities},  
volume = 46,  
year = 1993,  
number = 2,  
pages = {44--57}  
}
```

A Question to Solve

```

\documentclass{report}
\usepackage{amsthm}
\newtheorem{mydef}{Definition}
\usepackage{makeidx}
\makeindex
\begin{document}
\begin{mydef}
Newtons's method \index{newton's Method} for the solution of  $f(x) = 0$  is defined by
\begin{equation}
x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}, \quad k=0,1,2,\ldots
\end{equation}
with prescribed starting value  $x_0$ . We implicitly assume in the defining formula
\end{mydef}
\nocite{*}
\bibliographystyle{plain}
\bibliography{bibfile}
\printindex

```

The Verse Environment

```
\begin{verse}
There is an environment
for verse \\
Whose features some poets
will curse.
```

```
For instead of making\\
Them do \emph{all} line breaking, \\
It allows them to put too many words
on a line when they'd rather be
forced to be terse.
\end{verse}
```

The Verse Environment

*There is an environment for verse
Whose features some poets will curse.*

*For instead of making
Them do all line breaking,
It allows them to put too many words on a line when they'd
rather be forced to be terse.*

Table of Contents

- 1 Introduction
- 2 Why Even Bother...
- 3 Show
- 4 A Basic Document
 - Writing L^AT_EXcode
 - Basic Formatting
- 5 L^AT_EX and You**
 - The Files
- 6 L^AT_EX Extended
 - Common Errors
 - Macros
- 7 Conclusion

The File Structure

In your IEEE directory, you will notice several files. . .

- `main.tex` brings everything together, don't edit it
- `preamble.tex` contains any additional packages or macros
- `cover.tex` contains the cover information (title, author, etc.)
- `abstract.tex` and `summary.tex` contain the text of your scientific abstract and executive summary, respectively
- `paper.tex` contains the main body of your paper, including any and all figures, tables, etc.
- `biblio.bib` is a BibT_EX file containing your references
- `appa.tex` contains the text of any appendices you may have

Compile using `make main.pdf`

The Title Page

cover.tex is where you define the content of your title page

- It includes declarations of the title, author, and date
- You should replace the title and author as needed, but leave the date alone

```
\title{Length-enhanced superlative verbiage}  
\author{Joe Everystudent  
\vspace{0.5in}\\  
under the direction of\\  
Dr. Famous Person\\  
University of Hyderabad  
\vspace{1in}}
```

- The title page is created automatically using the `maketitle` command in `main.tex`

Abstract and Summary

- Your final paper will have both a technical abstract and a non-technical summary
- All you need to do is fill in the text, and the template takes care of the rest

Behind the Scenes

```
\begin{abstract}  
\input{abstract}  
\vspace{1in}  
\begin{center}\textbf{Summary}\end{center}  
\input{summary}  
\end{abstract}
```

Bibliography

biblio.bib acts as a database of references, and only includes in the bibliography those references you cite in your paper

BibT_EX

```
@article{nameofentry,  
author = {John Backus},  
title = {Symmetric Encryption},  
journal = {Journal of Modalities},  
volume = 46,  
year = 1993,  
number = 2,  
pages = {44--57}  
}
```

The Paper

L^AT_EX is built off of the idea of structure over formatting

```
\section{Introduction}
```

Layers of sectioning

section

subsection

subsubsection

paragraph

subparagraph

Referencing

References

```
\section{Results}\label{res}
```

...

As seen in Section `\ref{res}` ...

Footnotes

```
...telephony\footnote{Phony telephones}
```

Citations

Redundancy `\cite{nameofentry}`

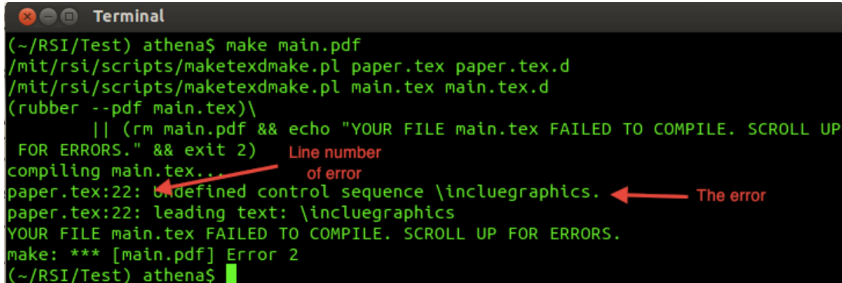
For multiple citations:

```
...methodology \cite{nameofentry, nameofotherentry}
```

Table of Contents

- 1 Introduction
- 2 Why Even Bother...
- 3 Show
- 4 A Basic Document
 - Writing L^AT_EXcode
 - Basic Formatting
- 5 L^AT_EX and You
 - The Files
- 6 L^AT_EX Extended
 - Common Errors
 - Macros
- 7 Conclusion

The Structure of an Error



A terminal window titled "Terminal" shows the output of a LaTeX compilation process. The user runs `make main.pdf` in the directory `~/RSI/Test`. The terminal displays the commands used to generate the PDF file, followed by an error message. Two red arrows point to specific parts of the error message: one points to the line number "22" and the other points to the text "The error".

```
(~/RSI/Test) athena$ make main.pdf
/mit/rsi/scripts/maketexdmake.pl paper.tex paper.tex.d
/mit/rsi/scripts/maketexdmake.pl main.tex main.tex.d
(rubber --pdf main.tex)\
    || (rm main.pdf && echo "YOUR FILE main.tex FAILED TO COMPILE. SCROLL UP
FOR ERRORS." && exit 2)
compiling main.tex...
paper.tex:22: undefined control sequence \incluegraphics.
paper.tex:22: leading text: \incluegraphics
YOUR FILE main.tex FAILED TO COMPILE. SCROLL UP FOR ERRORS.
make: *** [main.pdf] Error 2
(~/RSI/Test) athena$
```

Line number
of error

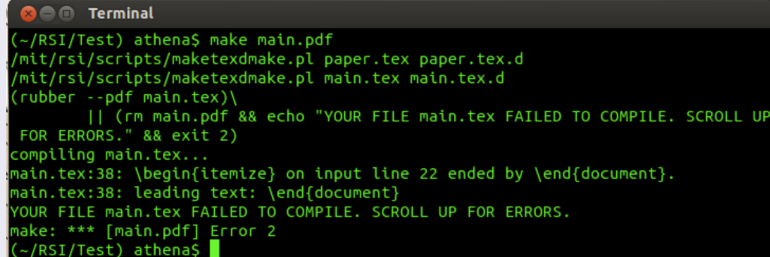
The error

Missing Environment End

The Code

```
\begin{itemize}  
\item Text.
```

The Error Message



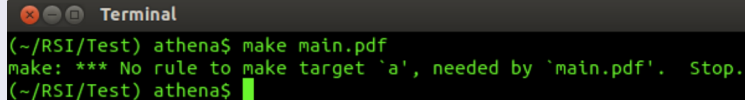
```
Terminal  
(~/RSI/Test) athena$ make main.pdf  
/mit/rsi/scripts/maketexdmake.pl paper.tex paper.tex.d  
/mit/rsi/scripts/maketexdmake.pl main.tex main.tex.d  
(rubber --pdf main.tex)\n    || (rm main.pdf && echo "YOUR FILE main.tex FAILED TO COMPILE. SCROLL UP  
FOR ERRORS." && exit 2)  
compiling main.tex...  
main.tex:38: \begin{itemize} on input line 22 ended by \end{document}.  
main.tex:38: leading text: \end{document}  
YOUR FILE main.tex FAILED TO COMPILE. SCROLL UP FOR ERRORS.  
make: *** [main.pdf] Error 2  
(~/RSI/Test) athena$
```


Spaces in Filenames

The Code

```
\includegraphics{a picture.png}
```

The Error Message

A terminal window titled "Terminal" with standard window controls (close, maximize, minimize). The terminal shows a command prompt at (~)/RSI/Test) athena\$ where the user has entered 'make main.pdf'. The output is an error message from 'make' stating that there is no rule to make target 'a', which is needed by 'main.pdf'. The prompt returns to (~)/RSI/Test) athena\$ with a cursor.

```
(~/RSI/Test) athena$ make main.pdf
make: *** No rule to make target `a', needed by `main.pdf'.  Stop.
(~/RSI/Test) athena$ █
```

Forgetting to Escape

The Code

```
a_b
```

The Error Message

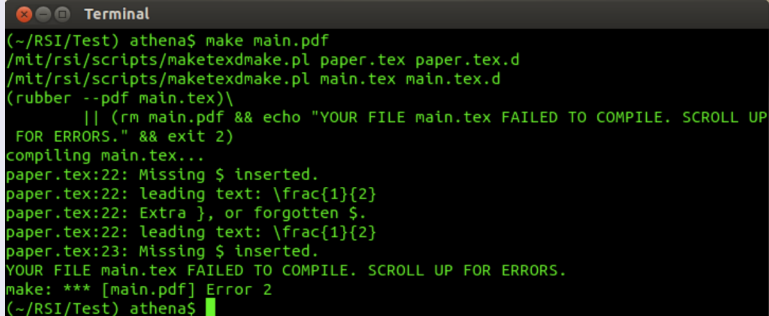
```
Terminal
(~/.RSI/Test) athena$ make main.pdf
/mit/rsi/scripts/maketexdmake.pl paper.tex paper.tex.d
/mit/rsi/scripts/maketexdmake.pl main.tex main.tex.d
(rubber --pdf main.tex)\
  || (rm main.pdf && echo "YOUR FILE main.tex FAILED TO COMPILE. SCROLL UP
FOR ERRORS." && exit 2)
compiling main.tex...
paper.tex:22: Missing $ inserted.
paper.tex:22: leading text: a_
paper.tex:23: Missing $ inserted.
YOUR FILE main.tex FAILED TO COMPILE. SCROLL UP FOR ERRORS.
make: *** [main.pdf] Error 2
(~/.RSI/Test) athena$
```

Forgetting to Use Math Mode

The Code

```
\frac{1}{2}
```

The Error Message

A terminal window titled "Terminal" with a dark background and green text. It shows the execution of a script to compile a LaTeX document. The script runs 'make main.pdf' and then 'rubber --pdf main.tex'. The compilation fails with several error messages from 'paper.tex' regarding missing dollar signs and extra closing braces for the fraction command. The terminal output is as follows:

```
(~/RSI/Test) athena$ make main.pdf
/mit/rsi/scripts/maketexdmake.pl paper.tex paper.tex.d
/mit/rsi/scripts/maketexdmake.pl main.tex main.tex.d
(rubber --pdf main.tex)\
  || (rm main.pdf && echo "YOUR FILE main.tex FAILED TO COMPILE. SCROLL UP
  FOR ERRORS." && exit 2)
compiling main.tex...
paper.tex:22: Missing $ inserted.
paper.tex:22: leading text: \frac{1}{2}
paper.tex:22: Extra }, or forgotten $.
paper.tex:22: leading text: \frac{1}{2}
paper.tex:23: Missing $ inserted.
YOUR FILE main.tex FAILED TO COMPILE. SCROLL UP FOR ERRORS.
make: *** [main.pdf] Error 2
(~/RSI/Test) athena$
```

Defining Theorems and More

The Code

```
% This is preamble.tex  
\newtheorem{name}{Display Name}
```

Example

```
%This is preamble.tex  
\newtheorem{thm}{Theorem}
```

Example, continued

```
% This is paper.tex  
\begin{thm}  
Herding cats is hard.  
\end{thm}
```

More on Theorems

Adding a Reference

```
\begin{thm}[Cain, 2002]  
Herding Rickoids is harder.  
\end{thm}
```

Proving your Theorems

```
% This is paper.tex  
\begin{proof}  
...  
\end{proof}
```

What are Macros?

- L^AT_EX allows you to define or redefine commands as you please
- In fact, L^AT_EX itself is a set of macros on top of T_EX

```
\newcommand{name}[num]{definition}
```

Resetting Commands

Changing lengths

```
\setlength{command}{length}
```

Ex.

```
\setlength{\parindent}{1cm}
```

```
\setlength{\parskip}{1cm plus4mm minus3mm}
```

Changing titles

Ex.

```
\renewcommand{\abstractname}{Summary}
```

Table of Contents

- 1 Introduction
- 2 Why Even Bother...
- 3 Show
- 4 A Basic Document
 - Writing L^AT_EXcode
 - Basic Formatting
- 5 L^AT_EXand You
 - The Files
- 6 L^AT_EXExtended
 - Common Errors
 - Macros
- 7 Conclusion**

So, why L^AT_EX?

- L^AT_EX allows you to worry about the content and the structure, rather than the presentation.
- L^AT_EX has one of the most advanced math typesetting systems around.
- L^AT_EX is incredibly extendible.
- L^AT_EX keeps track of references so you don't have to.
- L^AT_EX allows you to make more consistent, and more easily changeable, documents.

Getting Help and Learning More

- L^AT_EX Wikibooks:
en.wikibooks.org/wiki/LaTeX
- The Not So Short Introduction to L^AT_EX 2_ε :
www.ctan.org/tex-archive/info/lshort/english/lshort.pdf
- A Short Math Guide for L^AT_EX:
[ftp://ftp.ams.org/pub/tex/doc/amsmath/short-math-guide.pdf](http://ftp.ams.org/pub/tex/doc/amsmath/short-math-guide.pdf)
- The Beamer Theme Matrix:
www.hartwork.org/beamer-theme-matrix/

Google is still your best friend