



Basic Text Formatting with L^AT_EX

Wenting Deng

Department of Electrical Engineering

June 30th 2010



Special Characters

- Single quotation marks: left ‘ and right ’
- Double quotation marks: two single left quotes ‘ ‘ and two single right quotes ’ ’ or the double quote key (“ ”)
- A double quote followed by a single quote, or vice-versa: command `\`, between two quotation marks

For example: “*‘Fi’ or ‘fum’*”he asked



Special Characters

There are three kinds of dashes in typeset documents by typing one, two, or three “-” characters:

An intra-word dash, as in X-ray

An intra-word dash, as in X-ray

A medium dash for number range, like 1- -2

A medium dash for number range, like 1–2

A punctuation dash- - -like this

A punctuation dash—like this



Special Characters

<u>Character</u>	<u>Type in file</u>
#	<code>\#</code>
\$	<code>\\$</code>
%	<code>\%</code>
&	<code>\&</code>
-	<code>\-</code>
>	<code>\^{} </code>
~	<code>\~{} </code>
{	<code>\{</code>
}	<code>\}</code>
\	<code>\$\$\backslash\$</code>
<	<code>\$\$<\$ (or \textless)</code>
>	<code>\$\$>\$ (or \textgreater)</code>



Special Characters

Command	Output	Command	Output
$\backslash ' \{o\}$	ò	$\backslash ' \{o\}$	ó
$\backslash " \{o\}$	ö	$\backslash H \{o\}$	ő
$\backslash ^ \{o\}$	ô	$\backslash \sim \{o\}$	õ
$\backslash v \{o\}$	õ	$\backslash = \{o\}$	ō
$\backslash b \{o\}$	o	$\backslash . \{o\}$	ò
$\backslash d \{o\}$	o	$\backslash c \{o\}$	q
$\backslash r \{o\}$	o	$\backslash t \{oo\}$	öö

<code>\textit{...}</code>	<i>italic</i>	Italic shape, used mostly for emphasis
<code>\textsl{...}</code>	<i>slanted</i>	Slanted shape, a bit different from italic
<code>\textsc{...}</code>	SMALL CAPS	Small caps shape, use sparingly
<code>\textup{...}</code>	upright	Upright shape, usually the default
<code>\textbf{...}</code>	boldface	Boldface series, often used for headings
<code>\textmd{...}</code>	medium	Medium series, usually the default
<code>\textrm{...}</code>	roman	Roman family, usually the default
<code>\textsf{...}</code>	sans serif	Sans Serif family, used for posters, etc.
<code>\texttt{...}</code>	typewriter	Typewriter family, fixed-pitch characters
<code>\emph{...}</code>	<i>emphasized</i>	Use for emphasis, usually changes to italic

These commands can be combined, provided the font thus requested actually exists.

For example: `\textbf{\textit{This is bold italic}}`
produces: ***This is bold italic.***



Type Size

<code>\normalsize</code>	normal size	<code>\large</code>	large
<code>\small</code>	small	<code>\Large</code>	larger
<code>\footnotesize</code>	smaller than small	<code>\LARGE</code>	larger still
<code>\scriptsize</code>	smaller still	<code>\huge</code>	huge
<code>\tiny</code>	tiny	<code>\Huge</code>	hugest

- The actual size produced by each command depends on the initial point size selected for the document
- The size-changing commands are usually used within a group (i.e., braces) to delimit the range of their action
- To change both type size and style at the same time, commands can be used together



Sentences and Paragraphs

- T_EX ignores the way the input is formatted, but pay attention only to the logical concepts end of sentence and end of paragraphy.
- The command “\ ” will force a new line.
- A value inside square brackets following “\ ” will specify the amount of blank space between lines.
- The command to force a new page is \newpage.



Sentences and Paragraphs

Example:

The ends of words and sentences are marked by spaces
you type. It doesn't matter how many spaces you type;
one is as good as 100.

The ends of words and sentences are marked by spaces
you type. It doesn't matter how many spaces you type; one
is as good as 100.

This Line will be 20pt away from previous sentence.



Preventing Line Breaks

- Line breaking should be prevented at certain interword spaces.

For example, the expression “Chapter 3” looks strange if the “Chapter” ends one line and the “3” begins the next.

- A tilde character `~` produces an ordinary interword space at which T_EX will never break a line.

Mr. ~ Jones Mr. Jones
U. ~ S. ~ Grant U. S. Grant



Horizontal and Vertical Space

- If you want to leave horizontal and vertical space in your text, use commands `\hspace*` and `\vspace*`.
- If you want to center one line, use the command `{\centerline...}`

```
{\centerline This line will be centered.}
           This line will be centered.
```

```
\hspace*{1in} This text starts from a one-inch space.
                This text starts from a one-inch space.
```



Example of Enumerate

```
\begin{enumerate}  
  \item Sugar  
  \item Cream  
  \item Chocolate  
\end{enumerate}
```

1. Sugar
2. Cream
3. Chocolate

Example of Itemize

```
\begin{itemize}  
  \item Mix all ingredients together.  
  \item Boil until the thermometer  
    reaches 112  $^{\circ}\text{C}$ .  
  \item Stir and cool.  
\end{itemize}
```

- Mix all ingredients together.
- Boil until the thermometer reaches 112 °C.
- Stir and cool.



Example of Description

```
\begin{description}
  \item[dog] A loving animal that
    likes to sleep on the furniture.
  \item[cat] Aloof creature that can
    warm your feet on a winter's night
  \item[horse] Large animal, gives
    great rides. Eats a lot, luckily
    doesn't sleep on the furniture.
\end{description}
```

dog A loving animal that likes to
sleep on the furniture.

cat Aloof creature that can warm
your feet on a winter's night

horse Large animal, gives great rides.
Eats a lot, luckily doesn't sleep
on the furniture.



Example of nesting list environments

Here are some useful environments:

```
\begin{itemize}
\item center environment
\item quote environment
\item the three list environments:
  \begin{enumerate}
  \item enumerate (uses numbers)
  \item itemize (uses bullets)
  \item description (uses words)
  \end{enumerate}
\end{itemize}
```

Here are some useful environments:

- center environment
- quote environment
- the three list environments:
 1. enumerate (uses numbers)
 2. itemize (uses bullets)
 3. description (uses words)



Verbatim Text

- If you use the verbatim environment, everything input between the begin and end commands are processed as if by a typewriter. All spaces and new lines are reproduced as given. Any \LaTeX command will be ignored and handled as plain text.
- If you use the alltt package, it's almost the same as verbatim except it still process other \LaTeX commands inside the begin and end commands.



Verbatim Text

```
\begin{verbatim}
The verbatim environment
  simply reproduces every
  character you input,
including all s p a c e s!
\end{verbatim}
```

The verbatim environment
simply reproduces every
character you input,
including all s p a c e s!

```
\begin{alltt}
Verbatim extended with the ability
to use normal commands. Therefore, it
is possible to \emph{emphasize} words in
this environment, for example.
\end{alltt}
```

Verbatim extended with the ability
to use normal commands. Therefore, it
is possible to *emphasize* words in
this environment, for example.



Comments

- Special character `%`, that will comment out all the rest of the line after itself
- An environment called `comment` will comment out everything within itself.
(Require “`verbatim` package”)

```
This is another  
\begin{comment}  
rather stupid,  
but helpful  
\end{comment}  
example for embedding  
comments in your document.
```

This is another example for embedding comments in your document.



Thank you!