

L^AT_EX—*Beginner*¹

Emmanuel Paradis

June 27, 2008

Contents

1	T_EX and L^AT_EX	2
2	Why Use L^AT_EX?	3
3	Installing L^AT_EX	4
4	Compiling and File Formats	4
5	The Minimum to Start with L^AT_EX	5
6	The Preamble	6
6.1	The Document Classes	7
6.2	Packages	8
6.3	Babel	8
7	Preparing a L^AT_EX Document With Emacs	10
8	The Text	11
8.1	Fonts	12
8.2	Controlling Spaces	14

¹© 2008 Emmanuel Paradis

Permission is granted to make and distribute copies, either in part or in full and in any language, of this document on any support provided the above copyright notice is included in all copies. Permission is granted to translate this document, either in part or in full, in any language provided the above copyright notice is included.

8.3	Controlling Hyphenation	15
8.4	Accents	16
8.5	Special Characters	16
9	Sectioning	17
10	Environments	18
10.1	The environments <code>itemize</code> , <code>enumerate</code> , and <code>description</code> . .	18
10.2	The environment <code>tabular</code>	19
10.3	The environment <code>verbatim</code>	21
11	Mathematical Expressions	22
11.1	Main Mathematical Notations	23
11.2	Parentheses and Other Delimiters	24
11.3	Greek Letters	25
11.4	Controlling Spaces in Math Mode	26

1 \TeX and \LaTeX

\TeX : a computer language created by Donald Knuth¹ “intended for the creation of beautiful books—and especially for books that contain a lot of mathematics”.

\LaTeX : a set of macros in \TeX , originally written by Leslie Lamport, that “represents a balance between functionality and ease of use”. \LaTeX has predefined document classes (articles, slides, letters, books, . . .), as well as a large number of “packages” that ease the use of the functionalities of \TeX . \LaTeX keeps the features of the \TeX language (counters, variables, . . .).

Other systems derived from \TeX : \AMS-TeX , \AMS-LaTeX , \ConTeXt , \TeXinfo , \Eplain , \Lollipop .

The successors of \TeX : \Omega (Lambda), \e-TeX , \NTS .

¹www-cs-faculty.stanford.edu/~knuth/

2 Why Use \LaTeX ?

PROS

1. \LaTeX is available for all operating systems.
2. The documents made by \LaTeX are system-independant.
3. Numerous free distributions of \LaTeX are available.
4. The documents made by \LaTeX are of very good quality.
5. \LaTeX can make documents in PDF (Portable Document Format).
6. Numerous types of documents can be composed (books, articles, reports, theses, letters, slides, musical score-sheets, ...).
7. Mathematical expressions are printed with excellent quality.
8. The formats defined by default result in documents of excellent quality (typographically speaking) with little learning.
9. There are many base functionalities (counters, crossed references, ...) that are very useful to write a document (e.g., to number tables and figures).
10. Considering a document as a whole, rather than focusing on its aesthetic appearance on the screen, allows better results for its structure and display.
11. The formats and other functionalities (counters, environments, ...) can be modified (relatively) easily.
12. There is a very large number of “packages” which extend widely the possibilities of \LaTeX .
13. Various “linguistic environments” are available (languages and alphabets).
14. Many scientific publishers have \LaTeX templates available.

CONS

1. It is not possible to start a \LaTeX document without a minimum of knowledge.
2. \LaTeX is not WYSIWYG (what you see is what you get).²
3. Mastering \LaTeX requires progressive learning.

3 Installing \LaTeX

The installation of \LaTeX depends, of course, on the operating system.

Under Windows, the MiKTeX³ distribution is recommended; its installation is very easy thanks to a graphical interface. MiKTeX includes several executables and utilities (compilers, format converters, ...).

\LaTeX is available on any decent distribution of Linux.

MacTeX is a complete TeX system for MacOS X.⁴

4 Compiling and File Formats

A \LaTeX file is in ASCII format and includes the text together with the commands relative to its formatting. It is named with the extension `.tex`.

A \LaTeX file is compiled by TeX to create a DVI (*device independent*) file named with the extension `.dvi`. This file can be displayed using an appropriate program (*viewer*).

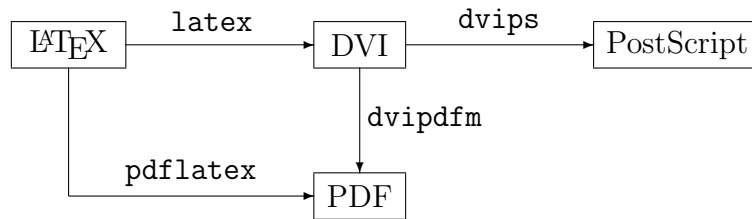
The compilation produces two other files: `.log` which contains a compilation report and possible error messages, and `.aux` (*auxiliary*) which is used by \LaTeX .

The distributions of \LaTeX generally include several executables for file format conversions.

²There are some graphical interfaces for \LaTeX particularly under Windows, most of them commercial.

³www.miktex.org

⁴www.tug.org/mactex



These executables are called in command line. For example, the present document has been created with:

```
pdflatex LaTeX_beginner_E.Paradis
```

Of course, it may be necessary to do some local configurations to make sure that the executable is accessible, e.g.:

```
c:\progra~1\miktex\miktex\bin\pdflatex LaTeX_debutant_E.Paradis
```

It is possible to write these commandes in a file `.bat` (for Windows) particularly if there are several of them like in the series of conversion shown above:

```
c:\progra~1\miktex\miktex\bin\latex LaTeX_debutant_E.Paradis.tex
c:\progra~1\miktex\miktex\bin\dvips LaTeX_debutant_E.Paradis.dvi
```

The use of one format or another depends on the user's specific needs. `pdflatex` creates directly PDF files from a `LATEX` document but it is incompatible with some packages that require a file in postscript format (like `PSTricks`). However, `pdflatex` is required by other packages (like `PPower4` to make slides).

5 The Minimum to Start with `LATEX`

To start with `LATEX` we need to:

- define the class of our document with `\documentclass{...}`, and
- declare the text with the commands `\begin{document}` and `\end{document}`.

A minimalist \LaTeX document may thus be:

```
\documentclass{article}
\begin{document}
Hello World!
\end{document}
```

This document may be compiled by \LaTeX to make, for instance, a DVI file.

However, in some tongues, like French, it is very useful to use the package `inputenc` with the option `latin1` which tells \LaTeX that the characters typed on an AZERTY keyboard are from the ISO-Latin 1 encoding. By default, \LaTeX assumes that the characters are coded with the ASCII encoding, and so does not recognize `é`, `è`, `ù`, `ç`, etc. Our document may thus become:

```
\documentclass{article}
\usepackage[latin1]{inputenc}
\begin{document}
Hello World!
\end{document}
```

6 The Preamble

The preamble of a \LaTeX document is the set of commands written before `\begin{document}`. It includes:

- the definition of the class of the document with its possible options;
- the declaration of the packages used in the document with the options if needed;
- any (re)definition of commands.

The preamble also contains the definition of the title and the author(s) of the document:

```
\title{...}  
\author{...}
```

A preamble may be very long (e.g., in templates).

6.1 The Document Classes

There are five default document classes in L^AT_EX: `article`, `report`, `letter`, `book`, and `slides`.

The declaration is made with:

```
\documentclass[options]{class}
```

where `options` specifies the options separated with commas. The following lists the most common options with the possible values (the letter `d` within parentheses indicates the default value).

- `a4paper`, `a5paper`, `b5paper`, `letterpaper` (`d`), `legalpaper`, `executivepaper`: size of paper.
- `landscape`: set format as landscape.
- `titlepage`, `notitlepage` (`d`): specifies whether there should be a separate title page.
- `leqno`: numbers equations on the left-hand side (right-hand side by default).
- `fleqn`: equations left-justified (centered by default).
- `draft`, `final` (`d`): if `draft` oversized text in the margins is indicated with a black ribbon.

The following options do not work with `slides`:

- `10pt` (d), `11pt`, `12pt`: base character size.
- `onecolumn` (d), `twocolumn`: number of columns.
- `oneside` (d), `twoside` (d for book): one-sided or two-sided.
- `openright` (d), `openany`: specifies whether a chapter must start on a left page.

The `slides` class has also the option `clock` to print the time at the bottom of each slide.

6.2 Packages

The packages eventually needed in a \LaTeX document are loaded in the preamble with the command `\usepackage[options]{package}`.

A \LaTeX installation includes several packages which description can be found in (with MiKTeX):

```
C:\Program Files\MiKTeX\doc\latex\base\usrguide.dvi
```

New packages are installed, for MiKTeX , with the MiKTeX Package Manager.

Each package has a specific function. We have already seen `inputenc`. We will see other examples later. The package `babel` has some special functionalities that we are going to see in detail in the next section.

6.3 Babel

Babel provides functionalities for formatting the text with respect to the tongue. This concerns general typography and hyphenation.

Note: `babel` is not a spell-checker.

The tongues used in a document must be declared as option of the document, or as option of the package. In the former case, the options are global and thus available to other packages.

The main command of `babel` is `\selectlanguage` which selects the tongue used to format the following text.

The preamble of the present document is:

```
\documentclass[12pt,french,english,a4]{article}
\usepackage[latin1]{inputenc}
\usepackage{babel}
\selectlanguage{english}
```

The command `\selectlanguage` may be used within the text.

The detailed documentation of `babel` is in (still with MiKTeX):

C:\Program Files\MiKTeX\doc\generic\Babel\babel.dvi

The following piece of text, written in French, illustrates that the same typed text will be formatted differently with respect to the tongue selected in `babel`.

Ce paragraphe a été composé le February 9, 2007: il a pour but d'illustrer certaines capacités de Babel. Dans la phrase précédente, j'ai utilisé la commande `\today` pour écrire la date. L^AT_EX a beaucoup de commandes similaires!

The same paragraph preceded by the command `\selectlanguage{french}` will give the following result:

Ce paragraphe a été composé le 9 février 2007 : il a pour but d'illustrer certaines capacités de Babel. Dans la phrase précédente, j'ai utilisé la commande `\today` pour écrire la date. L^AT_EX a beaucoup de commandes similaires!

There are six differences between these two paragraphs: see if you can spot them.

7 Preparing a L^AT_EX Document With Emacs

Emacs is a text editor with many functionalities, free, and available for many operating systems.

<http://www.gnu.org/software/emacs/>

Under Windows, we will use NTEmacs:

<http://www.gnu.org/software/emacs/windows/ntemacs.html>

Emacs comes with a package named RefTeX which facilitates the edition of L^AT_EX documents. We will also install the package AUCTeX which offers additional functionalities:

<http://www.gnu.org/software/auctex/>

These packages must be loaded when Emacs starts-up. We will thus edit the file `.emacs`. Here is an extract from my `.emacs` file showing a few useful customizations (semicolons indicate comments):

```
(show-paren-mode 1) ; parentheses highlighting
(global-font-lock-mode 1) ; syntax highlighting
(setq font-lock-maximum-decoration t) ; have it maximum!!
(setq frame-title-format "%b") ; put the name of the current
; buffer as the frame name
(setq column-number-mode t) ; display column number
;;; RefTeX

(add-hook 'LaTeX-mode-hook 'turn-on-reftex) ;with AUCTeX LaTeX mode
(add-hook 'latex-mode-hook 'turn-on-reftex) ;with Emacs latex mode

;; -----
;; AUC TeX

(add-to-list 'load-path "D:/auctex-10.0g/")
```

```
(add-to-list 'Info-default-directory-list "D:/auctex-10.0g/doc/")
(load "tex-site")
(require 'tex-site)
```

The file `.emacs` must be in the HOME directory.

Other useful applications: GhostScript and GSview to convert the formats PS, PDF, and others, and view these files:

<http://www.cs.wisc.edu/~ghost/>

8 The Text

In the vast majority of cases, the text starts with the command:

```
\maketitle
```

which will print the title, the author, and the date. To not print the date below the title, it can be redefined in the preamble avec `\date{}`.

\LaTeX will interpret the text between the commands `\begin{document}` and `\end{document}` as a series of commands in order to print the final document (DVI or PDF). There exist four types of commands:

command	meaning for \LaTeX
a character	print this character
space(s) or carriage-return	insert a space
blank line	new paragraph
<code>\command</code>	a command

The first, second, and third types of commands are executed with respect to some predefined, or defined by the user, parameters.

The two following texts will be treated in the same way by \LaTeX :

For LaTeX, there may be one or several spaces between the words.

For LaTeX, there may be one
or several
spaces between the words.

8.1 Fonts

By default, L^AT_EX uses the *Computer Modern Fonts* (CMF) created by D. Knuth. Three fonts are available: Roman, Sans Serif, and Typewriter.

These fonts exist under five shapes: upright, italics, bold, slanted, and small caps.

Other fonts may be used with an appropriate package, for instance, the package `times` to use Adobe fonts (Times, Helvetica, and Courier instead of Roman, Sans Serif, and Typewriter, respectively).

The control of fonts and shapes is done with two types of commands (e.g., to print text in italics):

- `\textit{...}` ou `\it`
- `{\it ...}`

The first type of commands will cumulate their effect with possible previous definitions, for example to result in bold italics; the second type of commands will ‘isolate’ the definition and the text will be simply in italics.

<code>\rm</code>	<code>\textrm{}</code>	Roman
<code>\sf</code>	<code>\textsf{}</code>	Sans Serif
<code>\tt</code>	<code>\texttt{}</code>	Typewriter
<code>\it</code>	<code>\textit{}</code>	<i>italics</i>
<code>\bf</code>	<code>\textbf{}</code>	bold
<code>\sl</code>	<code>\textsl{}</code>	<i>slanted</i> (\neq italics)
<code>\sc</code>	<code>\textsc{}</code>	SMALL CAPS
<code>\up</code>	<code>\textup{}</code>	upright

The combinations of these commands do not always work. It depends on the fonts installed. Here are a few examples which work with a base installation:

<code>\it\bfseries</code>	<i>bold italics</i>
<code>\sf\bfseries</code>	Sans Serif
<code>\sf\textit{}</code>	<i>Sans Serif italics</i>
<code>\sl\bfseries</code>	<i>bold slanted</i>
<code>\tt\textit{}</code>	<i>Typewriter italics</i>
<code>\tt\textsl{}</code>	<i>Typewriter slanted</i>

Note the commands to emphasize some text, `\textem{}` and `\em`, which by default print the text in italics.

The character size is controlled with commands modifying the size with respect to the base size defined in the document class (10 pt by default). The following table gives absolute sizes in points for each possible base size.

command	10 pt	11 pt	12 pt
<code>\tiny</code>	5	6	6
<code>\scriptsize</code>	7	8	8
<code>\footnotesize</code>	8	9	10
<code>\small</code>	9	10	11
<code>\normalsize</code>	10	11	12
<code>\large</code>	12	12	14
<code>\Large</code>	14	14	17
<code>\LARGE</code>	17	17	20
<code>\huge</code>	21	20	25
<code>\Huge</code>	25	25	25

Finally, to underline some text we shall use the special command `\underline`.

It is rare to see `\underline{underlined text}` in a printed document.

It is rare to see underlined text in a printed document.

Other effects (double-underlined, striked-out, wave-underlined, ...) are also available with other packages.

More details on fonts (control, definition, ...) may be found in:

C:\Program Files\MiKTeX\doc\latex\base\fntguide.dvi

8.2 Controlling Spaces

L^AT_EX controls spaces between words with respect to:

- parameters in action (format, selected tongue, ...),
- some characters.

By default, the text is left- and right-justified. Consequently, L^AT_EX spreads space between words so that all lines have the same width.

Of course, some spaces between words are substituted by linebreaks.

To make a space unbreakable, it can be replaced by a tilde that will link both words. It is sometimes preferable to avoid separating two words, here are a few examples:

Dr.~Jekyll Fig.~1 Table~2 D.~Knuth

Spaces that follow a command are ignored by L^AT_EX unless this command has curly braces. To avoid this behaviour, there are three possibilities:

<code>\LaTeX ignores spaces.</code>	L ^A T _E X ignores spaces.
<code>\LaTeX{} ignores spaces.</code>	L ^A T _E X ignores spaces.
<code>\LaTeX\ ignores spaces.</code>	L ^A T _E X ignores spaces.
<code>{\LaTeX} ignores spaces.</code>	L ^A T _E X ignores spaces.

The command ‘\ ’ (backslash followed by a space) adds explicitly a space but it is breakable (by contrast to the tilde).

In English, the text is justified by adding space unevenly between words and punctuation marks (see the above example). Particularly, a space after a full stop is longer than after a comma. However, by default L^AT_EX considers that if an uppercase letter is followed by a dot, this is an abbreviation and the following space is equal to one that follows a comma.

It is possible to alter this behaviour, in one direction or the other:

We programmed this in C\@. Denis et al.\ got similar results.

In French, a space is added before a double punctuation mark (made of two characters), even if this space has not been typed. The size of this space is adjusted by L^AT_EX if necessary, but it is unbreakable.

To cancel this space, we shall use the command `\string` right before the punctuation mark:

```
http://www...          http ://www...
http\string://www...   http://www...
```

8.3 Controlling Hyphenation

Hyphenation, which is done by default with L^AT_EX, eases the justification of the text. It depends on the selected tongue. There is no problem in English, but in French (and other tongues), it may be necessary to change the font encoding to allow L^AT_EX to hyphen the words correctly. This is done with the package `fontenc` and the option `T1` (to be placed in the preamble, of course):

```
\usepackage[T1]{fontenc}
```

If `babel` does not find where to hyphen some words, it is possible to indicate explicitly where this may be done:⁵

```
La gé\ -no\ -mi\ -que est en plein déve\ -lop\ -pe\ -ment.
```

It is also possible to prevent a word of being hyphaned:

```
\LaTeX\ is not a \mbox{WYSIWYG} system.
```

To avoid all hyphenation in the document, we shall add the following commands in the preamble (the text will remain right-justified):

```
\pretolerance=2000
\tolerance=3000
```

labelling right-justification generally cancels hyphenation.

⁵Again, this is mainly a problem in tongues other than English.

8.4 Accents

The command backslash followed by an accent means “add this accent on the letter that follows”.

<code>\‘e</code>	è	<code>\‘n</code>	ù	<code>\‘A</code>	À	<code>\‘p</code>	ù	<code>\‘{}</code>	˘
<code>\’e</code>	é	<code>\’n</code>	ú	<code>\’A</code>	Á	<code>\’p</code>	ú	<code>\’{}</code>	˙
<code>\^e</code>	ê	<code>\^n</code>	û	<code>\^A</code>	Â	<code>\^p</code>	û	<code>\^{}</code>	ˆ
<code>\"e</code>	ë	<code>\"n</code>	ÿ	<code>\"A</code>	Ä	<code>\"p</code>	ÿ	<code>\"{}</code>	¨
<code>\~e</code>	ẽ	<code>\~n</code>	ỹ	<code>\~A</code>	Ã	<code>\~p</code>	ỹ	<code>\~{}</code>	˜

We remark that the umlaut (¨) is obtained with double quotes ("), and not the corresponding key on AZERTY keyboards.

Recall that thanks to the the package `inputenc`, `ë` has the same meaning than `\"e`.

8.5 Special Characters

We have seen that some characters are used in the syntax of \LaTeX . They can be printed by being themselves preceded with a backslash. The latter is itself an exception since it has a special meaning when it is doubled.

<code>\textbackslash</code>	\	<code>\{</code>	{	<code>\}</code>	}
<code>\&</code>	&	<code>\#</code>	#	<code>\\$</code>	\$
<code>_</code>	-	<code>\%</code>	%		

We remark these other frequently used characters.

<code>\guillemotleft</code>	«	<code>--</code>	–	<code>\oe</code>	œ
<code>\guillemotright</code>	»	<code>---</code>	—	<code>\OE</code>	Œ
<code>\textdagger</code>	†	<code>“</code>	“	<code>\ae</code>	æ
<code>\textcopyright</code>	©	<code>”</code>	”	<code>\AE</code>	Æ
<code>\textregistered</code>	®	<code>‘</code>	‘	<code>\O</code>	Ø
<code>\textbullet</code>	•	<code>’</code>	’	<code>\o</code>	ø
<code>\dots</code>	...	<code>\c</code>	Ç	<code>\c</code>	Ç

In practice, in a French document we shall prefer the commands `\og` and `\fg` for guillemets.

9 Sectioning

A \LaTeX document may be sectioned into three hierarchical levels: sections, subsections, and subsubsections.

The corresponding commands are:

```
\section{...}  
  \subsection{...}  
    \subsubsection{...}
```

\LaTeX numbers these sections with Arabic numerals, and prints the headings with a font size proportional to the level. Numbering is done as follows:

- sections: 1, 2, ...
- subsections: 1.1, 1.2, ..., 2.1, 2.2, ...
- subsubsections: 1.1.1, 1.1.2, ..., 1.2.1, 1.2.2, ..., 2.1.1, 2.1.2, ...

The starred versions of these commands create the same levels of sectioning but without numbering:

```
\section*{...}  
  \subsection*{...}  
    \subsubsection*{...}
```

The paragraph and subparagraph levels are not numbered; the headings are just printed in bold:

```
\paragraph{...}  
  \subparagraph{...}
```

For the document classes `book` and `report`, two other levels are possible:

```
\part{...}  
  \chapter{...}
```

Parts are sets of chapters and are numbered with Roman numerals (I, II, ...). Chapters are numbered with Arabic numerals.

Whatever the class of the document, the table of contents may be printed with the command:

```
\tableofcontents
```

L^AT_EX will then create a file `.toc` containing the list of section headings. It may eventually be needed to compile the file `.tex` twice. The sections (and other subsections) that are not numbered will not be included in this table of contents.

10 Environments

An *environment* is a part of the document delimited by the commands `\begin{...}` and `\end{...}` between which the formatting parameters are modified. There exist many environments preprogrammed in L^AT_EX.

Emacs greatly facilitates the use of environments thanks to the menu ‘LaTeX’.

Environments have various functions: some simply modify the margins (`quote`), formatting (`center`, `flushright`, `flushleft`), or create a different environment with specific commands (`picture`).

In the next section, we will see the environments `equation` and `displaymath`.

10.1 The environments `itemize`, `enumerate`, and `description`

These three environments aim to display lists of items. The environments `itemize` and `enumerate` have the same syntax:

```
\begin{...}  
\item article 1  
\item article 2  
...  
\end{...}
```

There may be only one `\item`. The display differs in function of the environment:

- article 1
 - article 2
1. article 1
 2. article 2

It is noteworthy that the formatting of `itemize` depends on the selected tongue. The same list formatted in French will be:

- article 1
- article 2

Similarly for `enumerate`, the items are slightly less spaced in French than in English.

For the environment `description`, an optional heading within square brackets follows the command `\item`:

```
\begin{description}
\item[itemize:] items are simply listed;
\item[enumerate:] items are listed and numbered;
\item[description:] items are listed with a bold heading.
\end{description}
```

itemize: items are simply listed;

enumerate: items are listed and numbered;

description: items are listed with a bold heading.

10.2 The environment `tabular`

The environment `tabular` is frequently used in \LaTeX because it allows to create tables.

The command `\begin{tabular}` has a mandatory argument which defines the formatting of columns; it is made with a series of letters "c", "l" or "r" to center, left-justify, or right-justify, respectively.

Columns are separated with the character `&`.

Each row must end with an explicit linebreak indicated by a double backslash. This linebreak may occur even if the number of columns defined is not reached.

```
\begin{tabular}{lrc}
Species & Mass (g) & Scientific name\\
Mouse & 10 & {\it Mus musculus}\\
Whale & 120~000~000 & {\it Balaenoptera musculus}\\
Man & 70~000 & {\it Homo sapiens}\\
\end{tabular}
```

Species	Mass (g)	Scientific name
Mouse	10	<i>Mus musculus</i>
Whale	120 000 000	<i>Balaenoptera musculus</i>
Man	70 000	<i>Homo sapiens</i>

It is usual, and often more appealing aesthetically, to include a table within an environment `center`.

Horizontal lines used to separate the different parts of a table (i.e., rulers) are added with the command `\hline`:

```
\begin{center}
\begin{tabular}{lrc}
\hline
Species & Mass (g) & Scientific name\\
\hline
Mouse & 10 & {\it Mus musculus}\\
Whale & 120~000~000 & {\it Balaenoptera musculus}\\
Man & 70~000 & {\it Homo sapiens}\\
\hline
\end{tabular}
\end{center}
```

Species	Mass (g)	Scientific name
Mouse	10	<i>Mus musculus</i>
Whale	120 000 000	<i>Balaenoptera musculus</i>
Man	70 000	<i>Homo sapiens</i>

Vertical lines are added by specifying them in the column format definition with the symbol ‘|’:

```

\begin{center}
\begin{tabular}{|l|r|c|}
\hline
Species & Mass (g) & Scientific name\\
\hline
Mouse & 10 & {\it Mus musculus}\\
Whale & 120~000~000 & {\it Balaenoptera musculus}\\
Man & 70~000 & {\it Homo sapiens}\\
\hline
\end{tabular}
\end{center}

```

Species	Mass (g)	Scientific name
Mouse	10	<i>Mus musculus</i>
Whale	120 000 000	<i>Balaenoptera musculus</i>
Man	70 000	<i>Homo sapiens</i>

10.3 The environment `verbatim`

LaTeX prints characters included inside an environment `verbatim` as they are, without any formatting or interpreting eventual commands, using the current Typewriter font.

There are numerous examples of the use of this environment in this document.

The starred version of this environment prints also the spaces:

```

\begin{verbatim*}
\LaTeX\ ignores additional spaces.
\end{verbatim*}

```

`\LaTeX\ignorespaces\additionalspaces`.

The command `\verb` allows to include characters as they are within the text. The characters under consideration are delimited between two identical characters which are chosen by the user (except letters, *, and spaces).

the command <code>\verb \LaTeX </code> , <code>\dots</code>	the command <code>\LaTeX</code> , ...
the command <code>\verb+\LaTeX+</code> , <code>\dots</code>	the command <code>\LaTeX</code> , ...
the command <code>\verb5\LaTeX5</code> , <code>\dots</code>	the command <code>\LaTeX</code> , ...

11 Mathematical Expressions

There are two ways to include mathematical expressions in a document:

- Within a paragraph:
 - between `\(` and `\)`;
 - between two `$`;
 - in an environment `math`.
- On separate lines and centered (by default):
 - between `\[` and `\]`;
 - in an environment `displaymath`;
 - in an environment `equation`.

The environment `equation` numbers the equations sequentially.

In all these situations, L^AT_EX is said to be in *math mode* (by opposition to *text mode*).

The math mode has its own commands (which do not work in text mode!) and its own formatting rules.

1. Letters are considered as names of mathematical variables, and so are printed in italics. To print text upright, we shall use the commands `\textrm{}` or `\mathrm{}`.

2. Spaces are ignored. L^AT_EX adds space automatically around mathematical symbols ($=$, $+$, $-$, \times , \dots). Spaces can be controlled with specific commands.
3. Blank lines are disallowed.
4. The characters \wedge and $_$ are used to define exponents and indices, respectively.

We will take care that \wedge and $_$ consider only the single character that follows. If one wants to include several characters into the exponent or the index, they must be included within curly braces.

$$\text{\$x_ij\$} \quad x_{ij} \qquad \text{\$x_{ij}\$} \quad x_{ij}$$

11.1 Main Mathematical Notations

<code>\sqrt{x}</code>	\sqrt{x}	<code>\overbrace{u+v+w}</code>	$\overbrace{u+v+w}$
<code>\sqrt[n]{x}</code>	$\sqrt[n]{x}$	<code>\overrightarrow{AB}</code>	\overrightarrow{AB}
<code>\surd(x^2 + y^2)</code>	$\sqrt{x^2 + y^2}$	<code>\overleftarrow{CDE}</code>	\overleftarrow{CDE}
<code>\overline{x+y}</code>	$\overline{x+y}$	<code>\underline{x+y+z}</code>	$\underline{x+y+z}$
<code>x', x''</code>	x', x''	<code>\underbrace{f+g}</code>	$\underbrace{f+g}$
<code>\bar{x}</code>	\bar{x}	<code>></code>	$>$
<code>\dot{x}</code>	\dot{x}	<code><</code>	$<$
<code>\ddot{x}</code>	\ddot{x}	<code>\le</code>	\leq
<code>\hat{r}</code>	\hat{r}	<code>\ge</code>	\geq
<code>\widehat{r}</code>	\widehat{r}	<code>\equiv</code>	\equiv
<code>\tilde{r}</code>	\tilde{r}	<code>\sim</code>	\sim
<code>\widetilde{r}</code>	\widetilde{r}	<code>\simeq</code>	\simeq
		<code>\approx</code>	\approx
		<code>\neq</code>	\neq
		<code>\subset</code>	\subset
		<code>\in</code>	\in
		<code>\notin</code>	\notin
		<code>\mid</code>	$ $
		<code>\propto</code>	\propto

<code>\pm</code>	±	<code>\Leftarrow</code>	⇐	<code>\dots</code>	...
<code>\times</code>	×	<code>\rightarrow</code>	⇒	<code>\cdots</code>	...
<code>\div</code>	÷	<code>\leftrightarrow</code>	↔	<code>\vdots</code>	⋮
<code>\otimes</code>	⊗	<code>\Leftrightarrow</code>	⇔	<code>\ddots</code>	⋱
<code>\leftarrow</code>	←	<code>\uparrow</code>	↑	<code>\partial</code>	∂
<code>\rightarrow</code>	→	<code>\downarrow</code>	↓	<code>\infty</code>	∞

$$\{p \text{ \choose } n\} \binom{p}{n} \quad \frac{1}{x^2 + y^2} \quad \frac{1}{x^2 + y^2}$$

Most mathematical functions (which must be printed in Roman by contrast to variables) are typed simply preceded by a backslash.

`\cos \sin \exp \ln \log \max \Pr ...`

Integrals, sums, and products are typed with `\int`, `\sum`, and `\prod`. The lower and upper bounds are indicated as indices and exponents.

<code>\int</code>	f	<code>\sum</code>	Σ	<code>\prod</code>	Π
<code>\int_{x'}^{x''}</code>	$f(u) \mathrm{d}u$	<code>\sum_{i=1}^n</code>	x_i	<code>\prod_{i=1}^n</code>	$\Pr(x_i = X)$

11.2 Parentheses and Other Delimiters

Parentheses, braces, and other brackets are called *delimiters* in L^AT_EX's jargon. The size of these symbols are adjustable in left–right pairs.

The size of delimiters may be adjusted to fit the height of the expressions they enclose by preceding them with the commands `\left` and `\right`.

$$\begin{array}{ll} [\frac{1}{x}-(y+z)]^{1/2} & [\frac{1}{x}-(y+z)]^{1/2} \\ \left[\frac{1}{x}-(y+z)\right]^{1/2} & \left[\frac{1}{x}-(y+z)\right]^{1/2} \end{array}$$

It is possible to control the size of delimiters with the commands `\big`, `\Big`, `\bigg` and `\Bigg`.

$$\Bigg\{\bigg\{\Big\{\big\{\{$$

The delimiters available in \LaTeX are listed below.

<code>(</code>	<code>(</code>	<code>\langle</code>	<code><</code>	<code>\uparrow</code>	<code>↑</code>
<code>)</code>	<code>)</code>	<code>\rangle</code>	<code>></code>	<code>\downarrow</code>	<code>↓</code>
<code>[</code>	<code>[</code>	<code>\lfloor</code>	<code>⌊</code>	<code>\updownarrow</code>	<code>↕</code>
<code>]</code>	<code>]</code>	<code>\rfloor</code>	<code>⌋</code>	<code>\Uparrow</code>	<code>↑↑</code>
<code>\{</code>	<code>{</code>	<code>\lceil</code>	<code>⌈</code>	<code>\Downarrow</code>	<code>↓↓</code>
<code>\}</code>	<code>}</code>	<code>\rceil</code>	<code>⌉</code>	<code>\Updownarrow</code>	<code>↕↕</code>
<code>/</code>	<code>/</code>	<code>\backslash</code>	<code>\</code>		

11.3 Greek Letters

Lowercase Greek letters are typed with a command named from the letter under consideration.

$$\backslash\alpha \alpha \quad \backslash\beta \beta \quad \backslash\chi \chi \dots$$

For uppercase letters, there are two possibilities:

- The uppercase letter is identical to a Roman letter (alpha, beta, epsilon, ...), in which case the letter is used (A , B , E , ...);
- If not the case, the command is similar to the lowercase letter, but the first letter is a capital.

The list below gives the letters which fall in the second category.

<code>\Gamma</code>	Γ	<code>\Lambda</code>	Λ	<code>\Sigma</code>	Σ	<code>\Psi</code>	Ψ
<code>\Delta</code>	Δ	<code>\Xi</code>	Ξ	<code>\Upsilon</code>	Υ	<code>\Omega</code>	Ω
<code>\Theta</code>	Θ	<code>\Pi</code>	Π	<code>\Psi</code>	Ψ		

11.4 Controlling Spaces in Math Mode

We have seen at the beginning of this section that mathematical expressions may be placed either within a paragraph or on separate lines. For the expressions that need some space (e.g., with integrals), it is better to use the second solution. If such expressions are typed within a paragraph the formatting will be more compact.

the mean is `\frac{1}{n}\sum_{i=1}^n x_i`
the mean is $\frac{1}{n} \sum_{i=1}^n x_i$

It is possible to force such expressions to be printed in a classical way with the command `\displaystyle` that will be typed before the expression.

the mean is `\displaystyle \frac{1}{n}\sum_{i=1}^n x_i`
the mean is $\frac{1}{n} \sum_{i=1}^n x_i$

Horizontal spaces are defined by default in math mode. They can be controlled with special commands.

<code>x y</code>	$x y$	<code>x\; y</code>	$x y$	<code>x\quad y</code>	$x y$
<code>x\, y</code>	$x y$	<code>x\ y</code>	$x y$	<code>x\qqquad y</code>	$x y$
<code>x\: y</code>	$x y$				

The command `\!` shrinks the spaces. It can be repeated several times.

<code>x+y</code>	$x + y$	<code>x\!+\!y</code>	$x + y$	<code>x\!\!\!+\!\!\!y</code>	$x + y$
------------------	---------	----------------------	---------	------------------------------	---------