

L^AT_EX

Markus Kuhn



CTAN lion drawing by Duane Bibby

Computer Laboratory, University of Cambridge

<https://www.cl.cam.ac.uk/teaching/1617/TeX+MATLAB/>

Michaelmas 2016 – Part II

L^AT_EX – a document formatter

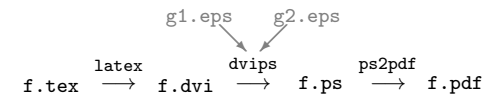
L^AT_EX: a macro package for T_EX (“tech”) typesetting system.

- ▶ excellent facilities for mathematical notation
- ▶ de-facto standard for preparing scientific publications in mathematical, physical, computing and engineering disciplines

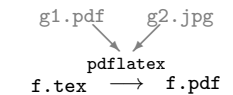
History: Donald Knuth (Stanford CS) developed T_EX in mid 1970s in SAIL to typeset his “Art of Computer Programming” books, reimplemented it in Pascal in the mid 1980s (WEB, literate programming), was later ported to C. Leslie Lamport (SRI, DEC) wrote L^AT_EX in early 1980s.

Both now community maintained as *T_EX Live* open-source distribution.

Classic processing steps:



Modern alternative:



These slides: prepared using the L^AT_EX beamer class.

L^AT_EX – resources

Recommended introduction

Leslie Lamport: L^AT_EX – a document preparation system. 2nd ed., Addison-Wesley, 1994.

(copies in CL library)

Installation

Debian Linux: `apt-get install texlive`
 macOS: <http://www.tug.org/mactex/>
 Windows: <http://www.tug.org/protext/>

More documentation

Online tutorials:

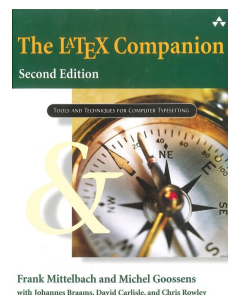
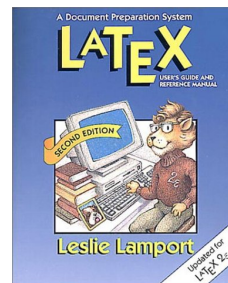
<http://www.latex-project.org/guides/>

T_EX Frequently Asked Questions list:

<http://www.tex.ac.uk/>

For advanced users:

Mittelbach, et al.: *The L^AT_EX Companion*. 2nd ed., Addison-Wesley, 2004.

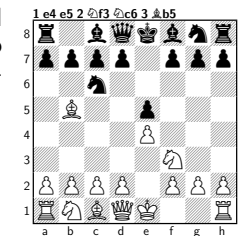


L^AT_EX – features and benefits

- ▶ Most popular mathematical typesetting language (subset imitations now also in: Word, MathJax, MediaWiki, etc.)
- ▶ Encourages logical markup ⇒ helps to maintain consistent style
- ▶ Plain-text source ⇒ easy to collaborate via version-control systems
- ▶ Command-line tool ⇒ easy to automate build (make)
- ▶ Use any plain-text editor you like (Emacs, vi, TeXworks, Word, etc.)
- ▶ Robust for large, complex documents (PhD thesis, books, etc.)
- ▶ Highly extensible (a Turing-complete macro programming language) ⇒ vast collection of add-on packages for special typesetting needs (figures, logic proofs, pseudo code, circuit diagrams, flow charts, chemical formulae, slides, chess positions, etc.)

```

\usepackage[skaknew]{skak}
\newgame
\mainline{1.e4 e5 2. Nf3 Nc6 3. Bb5}
\showboard
  
```



- ▶ Computer Modern, etc. ⇒ free font families
- ▶ Mature, free, portable, open source, used by many science publishers

L^AT_EX example

```
\documentclass[12pt]{article}
\setlength{\textwidth}{75mm}
\begin{document}
\title{\TeX -- a summary}
\author{Markus G. Kuhn}
\date{31 October 2016}
\maketitle
\thispagestyle{empty}

\section{Introduction}

Mathematical formul\ae\ such as
 $\$e^{i\pi} = -1$  or even
 $\left[ \Phi(z) = \frac{1}{\sqrt{2\pi}} \int_0^x e^{-\frac{1}{2}x^2} \right]$ 
were a real 'pain' to typeset until
\textsc{Knuth}'s text formatter \TeX
became available \cite{Knuth86}.

\begin{thebibliography}{9}
\bibitem{Knuth86}Donald E. Knuth:
The \TeX book. Ad\dison-Wesley, 1986.
\end{thebibliography}

\end{document}
```

T_EX – a summary

Markus G. Kuhn

31 October 2016

1 Introduction

Mathematical formulæ such as $e^{i\pi} = -1$ or even

$$\Phi(z) = \frac{1}{\sqrt{2\pi}} \int_0^x e^{-\frac{1}{2}x^2}$$

were a real 'pain' to typeset until KNUTH's text formatter T_EX became available [1].

References

[1] Donald E. Knuth: The T_EXbook. Ad-dison-Wesley, 1986.

5

T_EX input syntax

- ▶ T_EX reads plain-text *.tex files (e.g., prepared with emacs)
- ▶ no distinction is made between space character and line feed
- ▶ multiple spaces are treated like a single space
- ▶ multiple line feeds (empty lines) are treated as a paragraph separator (just like the \par command)
- ▶ command, macro and variable names start with a backslash (\), followed by either a sequence of letters or a single non-letter character (uppercase/lowercase is significant).
Correct: \par, \item, \pagethree, \LaTeX, \+, \\\, \3
Wrong: \page33, \<>
- ▶ space and line-feed characters are ignored if they follow a command/macro/variable name consisting of letters. Use _ to add an explicit space (e.g., \TeX\ syntax ⇒ T_EX syntax).

6

Characters with special semantics

In *.tex input files, the following are meta characters (part of syntax):

\$ % & ~ _ ^ \ { }

#\$%&_^{ } can be included in regular text using the macros

\# \\$ \% \& _ \^ \{ \}

Otherwise:

%	starts a comment	#	macro parameter
~	no-break space	&	tabulator mark
\$	delimits inline equations	\	macro prefix, \\ line separator
^	superscript	_	subscript (in math mode)
{	begin group	}	end group

Comments: All characters between (and including) a % and the next line feed will be ignored. Append % at the end of a line to avoid interpretation of the subsequent line feed as a space.

One could also insert \ and ~ as \textbackslash and \textasciitilde. But this is rarely ever done. The ASCII characters _{} are not typically used in regular text. They are common in computing-related strings (identifiers, source code, path names, URLs, etc.), for which it is customary to use a fixed-width typewriter font. There use the verbatim environment, or the macros \verb+...+ or \url{...}, which typeset all ASCII characters in typewriter font.

7

Grouping

The curly braces { and } serve two purposes in T_EX:

- ▶ Lexical scope: a { saves current state on a stack and } restores it, therefore state changes (e.g., font switch, variable assignment) inside a { ... } group last only until the }:

This is a {\bfseries bold} statement.

↓

This is a **bold** statement.

- ▶ Commands and macros read for each argument either a single character or a group enclosed by { and }:

Typeset \textbf M in \textbf{slanted style}.

↓

Typeset **M** in **boldface**.

L^AT_EX macros can have optional arguments, which are enclosed by [...]:

\makebox[80mm][c]{this is centered in a box 80 mm wide}

8

Changing font style manually

Declaring shape, series, family (each independently):

```
\mdseries Medium series \upshape Upright shape
\bfseries Boldface series \itshape Italic shape
\rmfamily Roman family \slshape Slanted shape
\sffamily Sans-serif family \scshape SMALL CAPS SHAPE
\ttfamily Typewriter family \normalfont Normal style
```

Corresponding commands that change the style only for their argument:

```
\textmd{text} \mdseries \textup{text} \upshape
\textbf{text} \bfseries \textit{text} \itshape
\textrm{text} \rmfamily \textsl{text} \slshape
\textsf{text} \sffamily \textsc{text} \scshape
\texttt{text} \ttfamily \textnormal{text} \normalfont
```

Change type size:

```
\tiny \small \large \huge
\scriptsize \normalsize \Large \Huge
\footnotesize \LARGE
```

For style consistency: use such low-level font commands only exceptionally (e.g., title pages, special notations). Otherwise leave any font adjustments to higher-level semantic commands such as `\emph{text}` or `\section{heading}`.

Typewriting versus typesetting

The ASCII (ISO 646) 7-bit character set has only 94 graphic characters:

```
!"#$%&'()*+,-./0123456789:;<=>?
@ABCDEFGHIJKLMNPQRSTUVWXYZ[\]^_
`abcdefghijklmnopqrstuvwxyz{|}~
```

They were chosen to cover the character repertoire of US typewriters and teletype printers. The committee added a few more symbols (`\``{}``|}``_`) in the hope that they will be useful for programming.

\TeX defines a number of shortcuts and macros to access the much larger set of “typographic” characters used by book printers.

These typographic characters still cannot be found on standard PC keyboard layouts, which were designed for 7-bit ASCII.

9

10

000	001	002	003	004	005	006	007	008	009	00A	00B	00C	00D	00E	00F	200	201	202	203	204	205	206	
0	[NU] [DL] [SP]	0	@	P	p			[XX] [DC] [NB]	o	À	Ð	à	ð			[ND] [SP]	-	†	%	o	⌘	[WJ]	
1	[SO] [DC]	!	1	A	Q	a	q	[XX] [PU]	i	±	Á	Ñ	á	ñ			[MO] [NB]	‡	‰	‰	‰	‰	‰
2	[ST] [DC]	"	2	B	R	b	r	[BP] [PU]	ç	²	Â	Ò	â	ò			[ES] [SP]	—	•	'	*	%	⌘
3	[ET] [DC]	#	3	C	S	c	s	[NB] [ST]	£	³	Ã	Ó	ã	ó			[EM] [SP]	—	▶	"	▪	~	⌘
4	[EOT] [DC]	\$	4	D	T	d	t	[IND] [CO]	¤	´	Ä	Ö	ä	ö			[SM] [SP]	—	.	'''	/	+	⌘
5	[ENG] [NAK]	%	5	E	U	e	u	[NEL] [MW]	¥	µ	Å	Ö	å	ö			[FM] [SP]	—	..	\	[*	⌘
6	[ACK] [SYN]	&	6	F	V	f	v	[SSA] [SPA]	¡	¶	Æ	Ø	æ	ø			[EM] [SP]		...	"]	:	[RL]
7	[BEL] [ETB]	'	7	G	W	g	w	[ESA] [EPA]	§	·	Ç	×	ç	÷			[E] [SP]	==	.	""	??	'''	[RL]
8	[BS] [CAN]	(8	H	X	h	x	[ES] [EPA]	¨	ˆ	È	Ø	è	ø			[P] [SP]	'	ˆ	?	!	ˆ	[RL]
9	[HT] [EM])	9	I	Y	i	y	[HTJ] [XXX]	©	¹	É	Û	é	ù			[P] [SP]	'	ˆ	?	!	ˆ	[RL]
A	[LF] [SUB]	*	:	J	Z	j	z	[VTS] [SCI]	ª	º	Ê	Û	ê	ú			[P] [SP]	'	ˆ	?	!	ˆ	[RL]
B	[VT] [ESC]	+	:	K	I	k	{	[PLD] [CSI]	«	»	Ë	Ü	ë	ü			[ZW] [SP]	'	ˆ	?	!	ˆ	[RL]
C	[FF] [FS]	<	L	\	l			[PLU] [ST]	¼	½	Ì	Ù	ì	ù			[C] [SP]	'	ˆ	?	!	ˆ	[RL]
D	[CR] [GS]	=	M	J	m	j	}	[R] [SHV]	½	¾	Í	Ý	í	ý			[Z] [SP]	'	ˆ	?	!	ˆ	[RL]
E	[SO] [RS]	>	N	^	n	~	}	[SS2] [PM]	¾	⅔	Î	Þ	î	þ			[L] [SP]	'	ˆ	?	!	ˆ	[RL]
F	[SI] [US]	/	?	O	_	o	DEL	[SSA] [APC]	—	¿	Ï	ß	ï	ÿ			[RL] [SP]	'	ˆ	?	!	ˆ	[RL]

The Unicode Standard 9.0, Unicode Inc., <http://www.unicode.org/charts/>

11

12

Dashes

ASCII provides only a single combined hyphen-minus character, but typesetters distinguish carefully between several dash characters:

```
- ⇒ - hyphen U+002D
-- ⇒ - en dash U+2013
--- ⇒ - em dash U+2014
$-$ ⇒ - minus U+2212
```

The hyphen (-) is the shortest of these and is used to combine separate words or split words across line-breaks.

The en dash (–) is often used to denote a range of numbers (as in pages 64–128), or – as in this example – as a punctuation dash.

The em dash is used—like this—as a punctuation dash, often without surrounding space, especially in US typography.

The minus (−) is a mathematical operator, whose shape matches the plus (+), unlike the hyphen or dashes. Compare: −+, −+, −+, −+.

Quotation marks

Typewriters and ASCII offer only unidirectional 'single' and "double" quotation marks, while typesetters use ‘curly’ and “directed” variants.

T_EX input files use the single quotation mark (') and the grave accent (̀) to encode these, as well the mathematical ‘prime’ marker and the French accents:

˘	⇒	‘	left quote	U+2018
'	⇒	’	right quote	U+2019
˘˘	⇒	“	left doublequote	U+201C
''	⇒	”	right doublequote	U+201D
\$' \$	⇒	'	prime	U+2032
\'u	⇒	ú	acute accent	U+00B4
\`u	⇒	ù	grave accent	U+0060

The apostrophe (it's) is identical to the right single quotation mark.

In some older terminal fonts (especially of US origin), the ˘ and ' characters have a compromise shape somewhere between the quotation marks ‘’ and the accents ´̀.

Non-ASCII Symbols

ı	!˘	Å	\AA	¶	\P
ı	?˘	ø	\o	†	\dag
œ	\oe	Ø	\O	‡	\ddag
Œ	\OE	†	\l	©	\copyright
æ	\ae	Ł	\L	£	\pounds
Æ	\AE	ß	\ss	...	\ldots
ä	\aa	§	\S		

Combining characters

ó	\'o	ō	\=o	ô	\t{o}
ò	\`o	ò	\.o	ø	\c{o}
ô	\^o	ö	\u{o}	ø	\d{o}
ö	\"o	ö	\v{o}	ø	\b{o}
õ	\~o	ő	\H{o}		

13

14

Space – the final frontier

Traditional English typesetting inserts a larger space at the end of a sentence. T_EX believes any space after a period terminates a sentence, unless it is preceded by an uppercase letter. Parenthesis are ignored.

This works often: J. F. Kennedy's U.S. budget. Look!

But not always: E.g. NASA. Dr. K. Smith et al. agree.

To correct failures of this heuristic, use

˘	⇒	no-break space
_	⇒	force normal space
\@	⇒	following punctuation ends sentence

as in

E.g. \ NASA \@. Dr.˘K. Smith et al.\ agree.

↓

E.g. NASA. Dr. K. Smith et al. agree.

Or disable the distinction of spaces with \frenchspacing.

Structure of a L^AT_EX document

First select a document class and its options, e.g. with

```
\documentclass[12pt,a4paper]{article}
```

Standard classes: article, report, book, letter, slides.

Publishers often provide authors with their own class as a *.cls file. Appendix A of *The L^AT_EX Companion* explains how to write new class files. A popular class for presentation slides: beamer

Environment: group delimited by \begin{name} and \end{name}, e.g.

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

Common environments: abstract, center, verbatim, itemize, enumerate, quote, tabular, equation, ...

Mark headings with

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

and L^AT_EX will take care of font sizes, numbering, and table of contents.

15

16

L^AT_EX list environments

```
\begin{itemize}
  \item Mammals
  \begin{itemize}
    \item Dogs
    \item Ponys
  \end{itemize}
  \item Insects
  \item[+] \ldots
\end{itemize}
```

- Mammals
 - Dogs
 - Ponys
- Insects
- + ...

```
\begin{enumerate}
  \setcounter{enumi}{-1}
  \item Fruits
  \begin{enumerate}
    \item Apples
    \item Cucumbers\label{c}
  \end{enumerate}
  \item Veggies
  \item \ldots
\end{enumerate}
```

- 0. Fruits
 - (a) Apples
 - (b) Cucumbers
- 1. Veggies (see also 0b)
- 2. ...

Package `enumitem` adds many configuration options to these environments. (`texdoc enumitem`)

17

Verbatim text: quoting source code

Lines of source code: the `verbatim` environment disables all meta characters, uses a typewriter font and preserves space and line feed:

```
\begin{verbatim}
$initial = substr($record->{'name'}, 0, 1);
\end{verbatim}
```

Do not indent a `verbatim` block (prints *all* whitespace). Keep `\end{verbatim}` on its own line.

Source code with math mode: `alltt` (`\usepackage{alltt}`) is similar to `verbatim`, but keeps the meta characters `\{}`, so you can still switch fonts and typeset mathematical expressions:

```
\begin{alltt}
for i := 1, ..., n
  print xi2
\end{alltt}
```

Inline strings: use `\verb+text+` to quote *text* inside a paragraph, where `+` is any character that does not occur in *text*. This also disables meta characters, preserves whitespace, and switches to a typewriter font.

You can't use `\verb` in command arguments, use `\texttt{text}` there instead.

The `\verb**text+` variant prints spaces as `␣`. Fix single quotation mark: `\usepackage{upquote}`.

18

Tweaking and extending L^AT_EX

L^AT_EX behaviour can be changed by overwriting predefined variables and macros. This can be done

- ▶ in the *preamble* (between the `\documentclass{...}` and `\begin{document}` lines) \Rightarrow for the entire document
- ▶ anywhere in the document \Rightarrow the effect will last only until the end of the current group (i.e., the next `}` or `\end{...}`)

Packages

A huge collection of extension packages exists for L^AT_EX. Some merely define additional macros and environments, others rewrite parts of L^AT_EX's internal machinery. For example, adding to the preamble

```
\usepackage{hyperref}
```

loads all the macros and settings defined in the `hyperref.sty` package.

`hyperref` adds new macros, such as `\url{...}` for typesetting URLs, but also automatically turns every reference to a page, section, or bibliographic entry into a hyperlink.

Documentation: `texdoc packagename` e.g. `texdoc geometry`

19

Example: changing page layout geometry

Adjust margins manually, via numerous length variables:

```
\setlength{\oddsidemargin}{-0.4mm} % 25 mm left margin
\setlength{\evensidemargin}{\oddsidemargin}
\setlength{\textwidth}{160mm} % 25 mm right margin
\setlength{\topmargin}{-5.4mm} % 20 mm top margin
\setlength{\headheight}{5mm}
\setlength{\headsep}{5mm}
\setlength{\footskip}{10mm}
\setlength{\textheight}{237mm} % 20 mm bottom margin
```

More comfortable:

```
\usepackage[vmargin=20mm,hmargin=25mm]{geometry}
```

The `geometry.sty` package automatically recalculates any dimensions not specified.

Make paragraphs not indented at the first line, but spaced apart slightly:

```
\setlength{\parindent}{0mm}
\setlength{\parskip}{\medskipamount}
```

Or just:

```
\usepackage{parskip}
```

20

Mathematical typesetting

In T_EX, mathematical formulas are formatted in a completely different mode from that used for normal text.

Inline formulas such as a_n (`a_n`) that appear as part of a normal paragraph have to be surrounded with `$...$` or `\(...\)`, while `\[...]` produces a displayed formula, such as

$$F_n = F_{n-1} + F_{n-2} \quad \backslash[F_n = F_{n-1} + F_{n-2} \backslash]$$

In math mode

- ▶ space characters are ignored; T_EX adds its own space around operators based on heuristics; manually add `thinspace` with “`\,`”
- ▶ a special *math italic* font is used, with different inter-character spacing, designed for *single-letter variables* concatenated in products
- ▶ many additional macros for special symbols are defined

Math italic is very *different* and not suitable for writing words or units!

Use `\mathrm{...}` around words, as in `v_diff` → v_{diff} .

Macros for common function symbols (constants: upright font!):

$$\max_{x \in \mathbb{R}} \{\log_2(\cos x)\} \quad \$\max_{x \in \mathbb{R}} \{\log_2(\cos x)\}$$$

Macros for neatly aligning multiple equations: `\usepackage{amsmath}`, see `texdoc amsldoc`.

21

Mathematical symbols

Greek letters

Γ	<code>\Gamma</code>	δ	<code>\delta</code>	π	<code>\pi</code>
Δ	<code>\Delta</code>	ϵ	<code>\epsilon</code>	ϖ	<code>\varpi</code>
Θ	<code>\Theta</code>	ε	<code>\varepsilon</code>	ρ	<code>\rho</code>
Λ	<code>\Lambda</code>	ζ	<code>\zeta</code>	ϱ	<code>\varrho</code>
Ξ	<code>\Xi</code>	η	<code>\eta</code>	σ	<code>\sigma</code>
Π	<code>\Pi</code>	θ	<code>\theta</code>	ς	<code>\varsigma</code>
Σ	<code>\Sigma</code>	ϑ	<code>\vartheta</code>	τ	<code>\tau</code>
Υ	<code>\Upsilon</code>	ι	<code>\iota</code>	υ	<code>\upsilon</code>
Φ	<code>\Phi</code>	κ	<code>\kappa</code>	ϕ	<code>\phi</code>
Ψ	<code>\Psi</code>	λ	<code>\lambda</code>	φ	<code>\varphi</code>
Ω	<code>\Omega</code>	μ	<code>\mu</code>	χ	<code>\chi</code>
α	<code>\alpha</code>	ν	<code>\nu</code>	ψ	<code>\psi</code>
β	<code>\beta</code>	ξ	<code>\xi</code>	ω	<code>\omega</code>
γ	<code>\gamma</code>	o	<code>o</code>		

22

Mathematical symbols

Binary operators

\pm	<code>\pm</code>	\triangleleft	<code>\lhd</code>	\oplus	<code>\oplus</code>
\mp	<code>\mp</code>	\cap	<code>\cap</code>	\ominus	<code>\ominus</code>
\setminus	<code>\setminus</code>	\cup	<code>\cup</code>	\otimes	<code>\otimes</code>
\cdot	<code>\cdot</code>	\uplus	<code>\uplus</code>	\oslash	<code>\oslash</code>
\times	<code>\times</code>	\sqcap	<code>\sqcap</code>	\odot	<code>\odot</code>
$*$	<code>\ast</code>	\sqcup	<code>\sqcup</code>	\dagger	<code>\dagger</code>
\star	<code>\star</code>	\wr	<code>\wr</code>	\ddagger	<code>\ddagger</code>
\diamond	<code>\diamond</code>	\bigcirc	<code>\bigcirc</code>	\amalg	<code>\amalg</code>
\circ	<code>\circ</code>	\triangleright	<code>\rhd</code>	\triangleleft	<code>\unlhd</code>
\bullet	<code>\bullet</code>	\vee	<code>\vee</code>	\triangleright	<code>\unrhd</code>
\div	<code>\div</code>	\wedge	<code>\wedge</code>		
\triangleleft	<code>\triangleleft</code>	\triangleup	<code>\bigtriangleup</code>		
\triangleright	<code>\triangleright</code>	∇	<code>\bigtriangledown</code>		

23

Mathematical symbols

Relations

\leq	<code>\leq</code>	\geq	<code>\geq</code>	\equiv	<code>\equiv</code>
\prec	<code>\prec</code>	\succ	<code>\succ</code>	\sim	<code>\sim</code>
\preceq	<code>\preceq</code>	\succeq	<code>\succeq</code>	\simeq	<code>\simeq</code>
\ll	<code>\ll</code>	\gg	<code>\gg</code>	\asymp	<code>\asymp</code>
\subset	<code>\subset</code>	\supset	<code>\supset</code>	\approx	<code>\approx</code>
\subseteq	<code>\subseteq</code>	\supseteq	<code>\supseteq</code>	\cong	<code>\cong</code>
\sqsubseteq	<code>\sqsubseteq</code>	\sqsupseteq	<code>\sqsupseteq</code>	\bowtie	<code>\bowtie</code>
\in	<code>\in</code>	\ni	<code>\ni</code>	\propto	<code>\propto</code>
\vdash	<code>\vdash</code>	\dashv	<code>\dashv</code>	\models	<code>\models</code>
\smile	<code>\smile</code>	\mid	<code>\mid</code>	\doteq	<code>\doteq</code>
\frown	<code>\frown</code>	\parallel	<code>\parallel</code>	\perp	<code>\perp</code>
\sqsubset	<code>\sqsubset</code>	\sqsupset	<code>\sqsupset</code>	\Join	<code>\Join</code>
$\not<$	<code>\not<</code>	$\not=$	<code>\not=</code>	$\not>$	<code>\not></code>
$\not\leq$	<code>\not\leq</code>	$\not\geq$	<code>\not\geq</code>	$\not\equiv$	<code>\not\equiv</code>
$\not\prec$	<code>\not\prec</code>	$\not\succ$	<code>\not\succ</code>	\dots	<code>\dots</code>

24

Mathematical symbols

Arrows

\leftarrow	<code>\leftarrow</code>	\Leftrightarrow	<code>\Leftrightarrow</code>
\Leftarrow	<code>\Leftarrow</code>	\longmapsto	<code>\longmapsto</code>
\rightarrow	<code>\rightarrow</code>	\hookrightarrow	<code>\hookrightarrow</code>
\Rightarrow	<code>\Rightarrow</code>	\rightharpoonup	<code>\rightharpoonup</code>
\leftrightarrow	<code>\leftrightarrow</code>	\rightharpoondown	<code>\rightharpoondown</code>
\Leftrightarrow	<code>\Leftrightarrow</code>	\leadsto	<code>\leadsto</code>
\mapsto	<code>\mapsto</code>	\uparrow	<code>\uparrow</code>
\hookleftarrow	<code>\hookleftarrow</code>	\Uparrow	<code>\Uparrow</code>
\leftharpoonup	<code>\leftharpoonup</code>	\downarrow	<code>\downarrow</code>
\leftharpoondown	<code>\leftharpoondown</code>	\Downarrow	<code>\Downarrow</code>
\rightleftharpoons	<code>\rightleftharpoons</code>	\updownarrow	<code>\updownarrow</code>
\longleftarrow	<code>\longleftarrow</code>	\Updownarrow	<code>\Updownarrow</code>
\Longleftarrow	<code>\Longleftarrow</code>	\nearrow	<code>\nearrow</code>
\longrightarrow	<code>\longrightarrow</code>	\searrow	<code>\searrow</code>
\Longrightarrow	<code>\Longrightarrow</code>	\swarrow	<code>\swarrow</code>
\longleftrightarrow	<code>\longleftrightarrow</code>	\nwarrow	<code>\nwarrow</code>

Mathematical symbols

\aleph	<code>\aleph</code>	\prime	<code>\prime</code>	\forall	<code>\forall</code>
\hbar	<code>\hbar</code>	\emptyset	<code>\emptyset</code>	\exists	<code>\exists</code>
\imath	<code>\imath</code>	∇	<code>\nabla</code>	\neg	<code>\neg</code>
\jmath	<code>\jmath</code>	\surd	<code>\surd</code>	\flat	<code>\flat</code>
ℓ	<code>\ell</code>	\top	<code>\top</code>	\natural	<code>\natural</code>
\wp	<code>\wp</code>	\perp	<code>\perp</code>	\sharp	<code>\sharp</code>
\Re	<code>\Re</code>	\parallel	<code>\parallel</code>	\clubsuit	<code>\clubsuit</code>
\Im	<code>\Im</code>	\angle	<code>\angle</code>	\diamondsuit	<code>\diamondsuit</code>
∂	<code>\partial</code>	\triangle	<code>\triangle</code>	\heartsuit	<code>\heartsuit</code>
∞	<code>\infty</code>	\backslash	<code>\backslash</code>	\spadesuit	<code>\spadesuit</code>
\Box	<code>\Box</code>	\diamond	<code>\Diamond</code>		

\dots	<code>\ldots</code>	\cdots	<code>\cdots</code>	\vdots	<code>\vdots</code>	\ddots	<code>\ddots</code>
---------	---------------------	----------	---------------------	----------	---------------------	----------	---------------------

Use `\ldots` in a, b, \dots, z , but `\cdots` in $a + b + \dots + z$. Never write ...

25

26

Large operators and delimiters

\sum	<code>\sum</code>	\bigcap	<code>\bigcap</code>	\bigodot	<code>\bigodot</code>
\prod	<code>\prod</code>	\bigcup	<code>\bigcup</code>	\bigotimes	<code>\bigotimes</code>
\coprod	<code>\coprod</code>	\bigsqcup	<code>\bigsqcup</code>	\bigoplus	<code>\bigoplus</code>
\int	<code>\int</code>	\bigvee	<code>\bigvee</code>	\biguplus	<code>\biguplus</code>
\oint	<code>\oint</code>	\bigwedge	<code>\bigwedge</code>		

These appear smaller inline than in displayed equations: \prod_0^{n-1} vs \prod_0^{n-1}

$[$	<code>\lbrack</code>	$]$	<code>\rbrack</code>
\lfloor	<code>\lfloor</code>	\rfloor	<code>\rfloor</code>
\lceil	<code>\lceil</code>	\rceil	<code>\rceil</code>
$\{$	<code>\lbrace</code>	$\}$	<code>\rbrace</code>
\langle	<code>\langle</code>	\rangle	<code>\rangle</code>
\llbracket	<code>\llbracket</code>	\rrbracket	<code>\rrbracket</code>
$\langle\langle$	<code>\langle\langle</code>	$\rangle\rangle$	<code>\rangle\rangle</code>

`\left(` (and `\right)`) grow delimiters to the height of what they enclose:

$$\left(\sum_{i=0}^{\infty} x^i \right) \quad \left(\sum_{i=0}^{\infty} x^i \right)$$

27

Mathematical symbols

Alternative names

\neq	<code>\ne</code>	$\{$	<code>\{</code>	\ni	<code>\owns</code>	$ $	<code>\vert</code>
\neq	<code>\neq</code>	$\}$	<code>\}</code>	\wedge	<code>\land</code>	$\ $	<code>\Vert</code>
\leq	<code>\le</code>	\rightarrow	<code>\to</code>	\vee	<code>\lor</code>		
\geq	<code>\ge</code>	\leftarrow	<code>\gets</code>	\neg	<code>\lnot</code>		

Stacking things

a^b	<code>a^{b}</code>	a_b	<code>a_{b}</code>
$\overline{a-b}$	<code>\overline{a-b}</code>	$\overbrace{a-b}$	<code>\overbrace{a-b}</code>
$\underline{a-b}$	<code>\underline{a-b}</code>	$\underbrace{a-b}_c$	<code>\underbrace{a-b}_c</code>

$$= \left\{ \begin{array}{l} a^{2^2}, \quad a \geq 0 \\ -\frac{1}{a}, \quad a < 0 \end{array} \right. = \left\{ \begin{array}{l} a^{\{2^{\{2\}}\}}, \quad \& a \geq 0 \\ -\frac{1}{\{a\}}, \quad \& a < 0 \end{array} \right.$$

28

Including graphics

DVI only supports characters and filled rectangles, but `dvips` and `pdftex` also understand embedded “special” instructions that provide more.

Embedded PostScript (EPS) vector graphics:

Normal PostScript files (*.ps) produce a sequence of pages. An EPS file describes only an image and is meant to be included into a PostScript page. EPS files lack instructions to output paper, but define a rectangular “bounding box”, using special `%%BoundingBox:` comments.

Load the `graphicx` extension of \LaTeX by adding

```
\usepackage{graphicx}
```

to the preamble. Then write

```
\includegraphics{filename.eps}
```

wherever you want to include the graphics file into your text.

In `pdflatex`, the `graphicx` package allows you to include graphics from PDF (vector graphics), JPEG (photos) and PNG (bitmap) files:

```
\includegraphics{filename.pdf}
```

29

Figures and references

Larger diagrams interfere with page breaking. They are best placed into a figure environment, such that \LaTeX can move them around. Example:

```
\begin{figure}
  \includegraphics[width=0.6\linewidth]{photo.jpg}
  \caption{This photograph shows the experimental setup.}
  \label{fig:expsetup}
\end{figure}
```

The automatically assigned figure number can be quoted as in:

```
See also Figure~\ref{fig:expsetup}
(page~\pageref{fig:expsetup}).
```

The `\label{...}` command can also be used after `\section{...}`, `\subsection{...}`, etc. and inside `\begin{equation} ... \end{equation}` to assign symbolic names to section and equation numbers, which can then be resolved via `\ref{...}` or `\pageref{...}`.

No need to manually renumber figures, sections, or equations!

31

Postscript/PDF graphics facilities

Applying coordinate transforms:

The `graphicx` package provides access to the geometric transform capabilities of the PostScript and PDF languages:

```
\scalebox{0.8}{\includegraphics{diagram.pdf}}
\includegraphics[height=60mm]{screenshot.png}
\includegraphics[width=0.9\linewidth]{photo.jpg}
\resizebox{190mm}{60mm}{becomes 19 cm $\times$ 6 cm large}
\resizebox{190mm}{!}{this becomes 19 cm wide}
\rotatebox{180}{this is upside down!}
```

Changing colours:

The `color` package also uses Postscript/PDF special commands:

```
This text is \textcolor{red}{printed in red} if ...
```

This text is **printed in red** if you include `\usepackage{color}`.

```
Default: \definecolor{red}{rgb}{1,0,0}
```

30

Build tools for \LaTeX

To make sure `\label` references and tables of contents use the correct numbers, it may be necessary to call `latex` twice. It will output “Rerun to get cross-references right” in this case.

The following implicit Makefile rule takes care of this:

```
.DELETE_ON_ERROR:
%.pdf %.aux %.idx: %.tex
    pdflatex $<
    while grep 'Rerun to get ' $*.log ; do pdflatex $< ; done
```

An alternative is the “`latexmk`” tool, which automatically determines dependencies (e.g. from `\includegraphics`) and recompiles \LaTeX documents where file modification timestamps indicate that this is necessary.

32

Graphics editor xfig

- ▶ Its *.fig files have a simple plain-text format that can be edited manually, script generated, and leads to useful diffs.
- ▶ Can export *.eps or *.pdf files
- ▶ Can also produce figures in which \LaTeX is used to fill in all the text. This provides math mode, macros, symbols, references, fonts that match the main text, etc.

Ask xfig to export a *.pstex + *.pstex_t file pair. The *.pstex file lacks the text parts of the figure. The *.pstex_t file contains \LaTeX commands that first load the *.pstex image, and then add all the text in the figure. Select the "special text" mode in xfig to enable \LaTeX metacharacters. Use `\include{*.pstex_t}` to add such a figure in your document. (PDF equivalent: *.pdftex + *.pdftex_t)

- ▶ Command-line export tool (e.g., for Makefile): fig2dev

```
%.eps: %.fig
    fig2dev -L eps $< $@
%.pstex %.pstex_t: %.fig
    fig2dev -L pstex_t -p $*.pstex $< $*.pstex_t
    fig2dev -L pstex $< $*.pstex
%.pdftex %.pdftex_t: %.fig
    fig2dev -L pdftex_t -p $*.pdftex $< $*.pdftex_t
    fig2dev -L pdftex $< $*.pdftex
```

Other graphics tools: TikZ, pnmtools, Inkscape, MATLAB, R, gnuplot, Python+matplotlib