

L^AT_EX

Markus Kuhn



CTAN lion drawing by Duane Bibby

Department of Computer Science and Technology
University of Cambridge

<https://www.cl.cam.ac.uk/teaching/2627/TeX+Julia/>

L^AT_EX – a document formatter

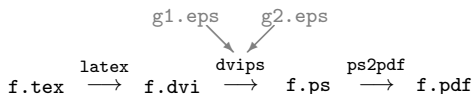
L^AT_EX: a macro package for the T_EX ($\tau\epsilon\chi$, “tech”) typesetting system

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

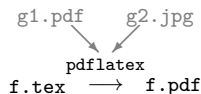
History: Donald Knuth (Stanford CS) developed T_EX in late 1970s, in SAIL, to typeset his “Art of Computer Programming” books, reimplemented it in Pascal in the early 1980s (WEB, literate programming), which was later ported to C. Leslie Lamport (SRI, DEC) wrote L^AT_EX in early 1980s.

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.

Recommended introduction

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

Installation

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

More documentation

Online tutorials:

<https://www.latex-project.org/help/>

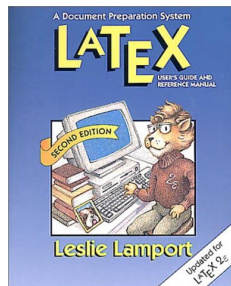
T_EX Frequently Asked Questions list:

<https://texfaq.org/>

For advanced users:

Mittelbach et al.: *The L^AT_EX Companion*.

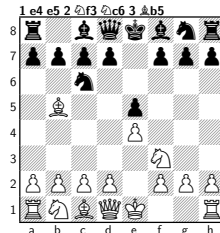
3rd ed., Addison-Wesley, 2023.



L^AT_EX – features and benefits

- ▶ Most popular mathematical typesetting language (subset imitations now also in: Word, MathJax, MediaWiki, etc.)
- ▶ Encourages logical markup $\Rightarrow$ helps to maintain consistent style
- ▶ Plain-text source $\Rightarrow$ easy to collaborate via version-control systems
- ▶ Command-line tool $\Rightarrow$ 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)
 $\Rightarrow$ 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. $\Rightarrow$ free font families
- ▶ Mature, free, portable, open source, used by many science publishers

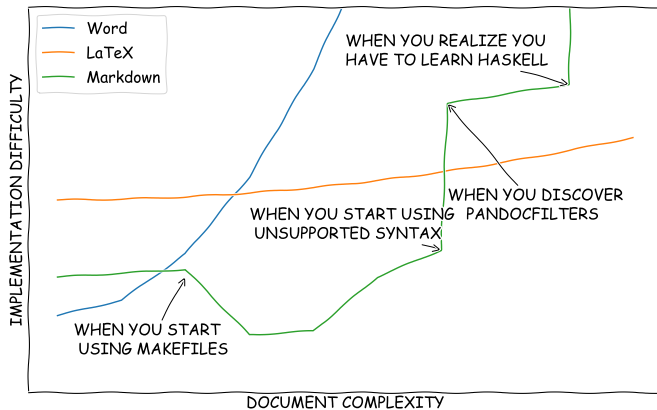


Figure: Dheepak Krishnamurthy, CC BY, <https://blog.kdheepak.com/writing-papers-with-markdown.html>

```

\documentclass[12pt]{article}
\setlength{\textwidth}{75mm}
\begin{document}
\title{\TeX\ -- a summary}
\author{Markus G. Kuhn}
\date{12 October 2026}
\maketitle
\thispagestyle{empty}

\section{Introduction}

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

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

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. Addison-Wesley, 1986.
\end{thebibliography}

\end{document}

```

T_EX – a summary

Markus G. Kuhn

12 October 2026

1 Introduction

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

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

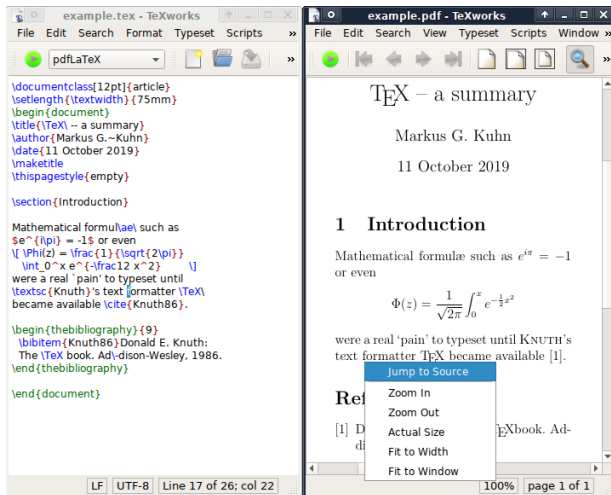
were a real ‘pain’ to typeset until KNUTH’s text formatter T_EX became available [1].

References

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

TeXworks GUI IDE

TeXworks is a $\text{\LaTeX}$ editor that shows source text and PDF output side by side and makes it easy to navigate between them:



There are even web-based $\text{\LaTeX}$ editors, e.g. Overleaf, Papeeria, Verbosus.

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
- ▶ empty lines act as 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 $\Rightarrow$ T_EX syntax).

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 hardly 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 macro \verb+...+, which typeset all ASCII characters in typewriter font, or \url{...} for URLs.

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{boldface}.



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}

Changing font style manually

Declaring shape, series, family (each independently):

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

Corresponding commands that change the style only for their argument:

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

Change type size:

<code>\tiny</code>	<code>\small</code>	<code>\large</code>	<code>\huge</code>
<code>\scriptsize</code>	<code>\normalsize</code>	<code>\Large</code>	<code>\Huge</code>
<code>\footnotesize</code>		<code>\LARGE</code>	

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:;<=>?  
@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_  
`abcdefghijklmnopqrstuvwxyz{|}~
```

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

T_EX 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.

	000	001	002	003	004	005	006	007
0	[NUL] 0000	[DLE] 0010	[SP] 0020	0	@	P	`	p
1	[SOH] 0001	[DC1] 0011	!	1	A	Q	a	q
2	[STX] 0002	[DC2] 0012	"	2	B	R	b	r
3	[ETX] 0003	[DC3] 0013	#	3	C	S	c	s
4	[EOT] 0004	[DC4] 0014	\$	4	D	T	d	t
5	[ENQ] 0005	[NAK] 0015	%	5	E	U	e	u
6	[ACK] 0006	[SYN] 0016	&	6	F	V	f	v
7	[BEL] 0007	[ETB] 0017	'	7	G	W	g	w
8	[BS] 0008	[CAN] 0018	(	8	H	X	h	x
9	[HT] 0009	[EM] 0019	)	9	I	Y	i	y
A	[LF] 000A	[SUB] 001A	*	:	J	Z	j	z
B	[VT] 000B	[ESC] 001B	+	;	K	[	k	{
C	[FF] 000C	[FS] 001C	,	<	L	\	l	
D	[CR] 000D	[GS] 001D	-	=	M	]	m	}
E	[SO] 000E	[RS] 001E	.	>	N	^	n	~
F	[SI] 000F	[US] 001F	/	?	O	_	o	[DEL] 001F

	008	009	00A	00B	00C	00D	00E	00F
0	[XXX] 0080	[DC5] 0090	[NB SP] 00A0	°	À	Đ	à	đ
1	[XXX] 0081	[PU1] 0091	ı	±	Á	Ñ	á	ñ
2	[BPH] 0082	[PU2] 0092	ç	²	Â	Ò	â	ò
3	[NBH] 0083	[STS] 0093	£	³	Ã	Ó	ã	ó
4	[IND] 0084	[CCH] 0094	α	´	Ä	Ô	ä	ô
5	[NEL] 0085	[MW] 0095	¥	μ	Å	Ö	å	ö
6	[SSA] 0086	[SPA] 0096	ı	¶	Æ	Ö	æ	ö
7	[ESA] 0087	[EPA] 0097	§	·	Ç	×	ç	÷
8	[HTS] 0088	[SOS] 0098	¨	¸	È	Ø	è	ø
9	[HTJ] 0089	[XXX] 0099	©	¹	É	Ù	é	ù
A	[VTS] 008A	[SCI] 009A		º	Ê	Ú	ê	ú
B	[PLD] 008B	[CSI] 009B	«	»	Ë	Û	ë	û
C	[PLU] 008C	[ST] 009C	¬	¼	Ì	Ü	ì	ü
D	[RI] 008D	[OSC] 009D	½	½	Í	Ý	í	ý
E	[SS2] 008E	[PM] 009E	®	¾	Î	Þ	î	þ
F	[SS3] 008F	[APC] 009F	—	¿	Ï	ß	ï	ÿ

	200	201	202	203	204	205
0	[NO SP] 2000	-	†	‰	—	◡
1	[MO SP] 2001	[NB SP] 2011	‡	‰	∧	*
2	[EN SP] 2002	—	•	'	***	%
3	[EM SP] 2003	—	►	"	■	~
4	[3/M SP] 2004	—	.	'''	/	—
5	[4/M SP] 2005	—	—	..	{	*
6	[6/M SP] 2006		...	"	}	:
7	[F SP] 2007	=	.	'''	??	'''
8	[P SP] 2008	‘	[SEP] 2018	^	?!	÷
9	[TH SP] 2009	’	[P SEP] 2019	<	!?	÷
A	[H SP] 200A	,	[LRE] 201A	>	7	:
B	[ZW SP] 200B	‘	[RLE] 201B	×	¶	÷
C	[ZW NJ] 200C	“	[PDF] 201C	!!	¶	÷
D	[ZW J] 200D	”	[LRO] 201D	!	¶	÷
E	[LRM] 200E	„	[RLO] 201E	—	*	÷
F	[RLM] 200F	“	[NNB SP] 201F	—	;	[MM SP] 205F

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 symmetric '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 as 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
å	\aa	§	\S		

Such characters can now also be entered as UTF-8 Unicode. (The setting previously required for this, `\usepackage[utf8]{inputenc}`, became the default in 2018.)

Combining characters

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

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. Parentheses 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

<code>~</code>	$\Rightarrow$	no-break space
<code>_</code>	$\Rightarrow$	force normal space
<code>\@</code>	$\Rightarrow$	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 $\text{\LaTeX}$ 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 $\text{\LaTeX}$ 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

<code>\section{...}</code>	<code>\subsubsection{...}</code>
<code>\subsection{...}</code>	<code>\paragraph{...}</code>

and $\text{\LaTeX}$ will take care of font sizes, numbering, and table of contents.

```
\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
    (see also \ref{c})
  \item \ldots
\end{enumerate}
```

0. Fruits

(a) Apples

(b) Cucumbers

1. Veggies (see also 0b)

2. ...

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,\ldots,n\ )
    print \ (x\sb{i}\sp{2}\ )
\end{alltt}
```

$$\text{for } i := 1, \dots, n$$
$$\text{print } x_i^2$$

Inline strings: use `\verb+text+` to quote *text* inside a paragraph, where + can be any non-letter 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}`.

Tweaking and extending L^AT_EX

L^AT_EX behaviour can be changed by overriding 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`

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}
```

Mathematical typesetting: *italic* vs upright letters

In mathematical typesetting, *italic type* is used for single letters denoting

- ▶ variables, such as x , y
- ▶ running numbers, such as i in $\sum_{i=0}^n x_i$
- ▶ parameters, such as a , b , which may be considered constant in a particular context
- ▶ arbitrary functions, such as f , g

whereas upright (roman) type is used for

- ▶ explicitly defined functions that do not depend on the context, such as $\sin$, $\exp$, $\ln$
- ▶ mathematical constants, the values of which never change, such as $e = 2.718281828\dots$, $i^2 = -1$
- ▶ well-defined operators such as div , d in $\frac{dx}{dt}$
- ▶ symbols for units of measurement, such as m , s , kHz , MW , $^{\circ}\operatorname{C}$

SI unit symbol capitalization: prefix mega and higher = uppercase letter; symbols for units named after a person start with an uppercase letter; all other letters are lowercase (litre $\operatorname{l/L}$).

Math mode

In $\text{T}_{\text{E}}\text{X}$, 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; $\text{T}_{\text{E}}\text{X}$ 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 `$v_\mathrm{diff}$` $\rightarrow v_{\text{diff}}$.

Macros for common function symbols:

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

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

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>	$\omicron$	<code>\omicron</code>		

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>	$\trianglelefteq$	<code>\unlhd</code>
$\bullet$	<code>\bullet</code>	$\vee$	<code>\vee</code>	$\trianglerighteq$	<code>\unrhd</code>
$\div$	<code>\div</code>	$\wedge$	<code>\wedge</code>		
$\triangleleft$	<code>\triangleleft</code>	$\triangle$	<code>\bigtriangleup</code>		
$\triangleright$	<code>\triangleright</code>	∇	<code>\bigtriangledown</code>		

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>
$\nless$	<code>\not<</code>	$\neq$	<code>\not=</code>	$\ngtr$	<code>\not></code>
$\nleq$	<code>\not\leq</code>	$\ngeq$	<code>\not\geq</code>	$\nequiv$	<code>\not\equiv</code>
$\nprec$	<code>\not\prec</code>	$\nsucc$	<code>\not\succ</code>	...	

Mathematical symbols

Arrows

$\leftarrow$	<code>\leftarrow</code>	$\longleftrightarrow$	<code>\Longleftarrow</code>
$\Lleftarrow$	<code>\Leftarrow</code>	$\mapsto$	<code>\longmapsto</code>
$\rightarrow$	<code>\rightarrow</code>	$\hookrightarrow$	<code>\hookrightarrow</code>
$\Rightarrow$	<code>\Rightarrow</code>	$\rightarrow$	<code>\rightarrow</code>
$\leftrightarrow$	<code>\leftrightarrow</code>	$\rightarrow$	<code>\rightarrow</code>
$\Leftrightarrow$	<code>\Leftrightarrow</code>	$\rightsquigarrow$	<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>
$\leftharpoonup$	<code>\leftharpoonup</code>	$\Downarrow$	<code>\Downarrow</code>
$\rightrightarrows$	<code>\rightrightarrows</code>	$\updownarrow$	<code>\updownarrow</code>
$\longrightarrow$	<code>\longrightarrow</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>	$'$	<code>\prime</code>	$\forall$	<code>\forall</code>	<code>\forall</code>	<code>\forall</code>
$\hbar$	<code>\hbar</code>	$\emptyset$	<code>\emptyset</code>	$\exists$	<code>\exists</code>	<code>\exists</code>	<code>\exists</code>
$\imath$	<code>\imath</code>	∇	<code>\nabla</code>	$\neg$	<code>\neg</code>	<code>\neg</code>	<code>\neg</code>
$\jmath$	<code>\jmath</code>	$\sqrt{}$	<code>\surd</code>	$\flat$	<code>\flat</code>	<code>\flat</code>	<code>\flat</code>
ℓ	<code>\ell</code>	$\top$	<code>\top</code>	$\natural$	<code>\natural</code>	<code>\natural</code>	<code>\natural</code>
$\wp$	<code>\wp</code>	$\bot$	<code>\bot</code>	$\sharp$	<code>\sharp</code>	<code>\sharp</code>	<code>\sharp</code>
$\Re$	<code>\Re</code>	$\parallel$	<code>\parallel</code>	$\clubsuit$	<code>\clubsuit</code>	<code>\clubsuit</code>	<code>\clubsuit</code>
$\Im$	<code>\Im</code>	$\angle$	<code>\angle</code>	$\diamondsuit$	<code>\diamondsuit</code>	<code>\diamondsuit</code>	<code>\diamondsuit</code>
∂	<code>\partial</code>	$\triangle$	<code>\triangle</code>	$\heartsuit$	<code>\heartsuit</code>	<code>\heartsuit</code>	<code>\heartsuit</code>
∞	<code>\infty</code>	$\backslash$	<code>\backslash</code>	$\spadesuit$	<code>\spadesuit</code>	<code>\spadesuit</code>	<code>\spadesuit</code>
$\Box$	<code>\Box</code>	$\Diamond$	<code>\Diamond</code>				

$\dots$ `\ldots` $\cdots$ `\cdots` $\vdots$ `\vdots` $\ddots$ `\ddots`

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

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:

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

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

$$a^b \quad \text{a}^{\{b\}}$$

$$a_b \quad \text{a}_{\{b\}}$$

$$\overline{a-b} \quad \text{\overline{a-b}}$$

$$\overbrace{a-b} \quad \text{\overbrace{a-b}}$$

$$\underline{a-b} \quad \text{\underline{a-b}}$$

$$\underbrace{a-b}_c \quad \text{\underbrace{a-b}_c}$$

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

Tables

```
\newcommand{\xor}
{\operatorname{\mathtt{xor}}}

\begin{tabular}{c|c|c}
$x$ & $y$ & $x \xor y$ \\ \hline
0 & 0 & 0 \\
0 & 1 & 1 \\
1 & 0 & 1 \\
1 & 1 & 0
\end{tabular}
```

x	y	$x \text{ xor } y$
0	0	0
0	1	1
1	0	1
1	1	0

The tabular environment supports these column types:

- `l` left-aligned lines
- `r` right-aligned lines
- `c` centered lines
- `|` vertical line (as column separator or border)
- `p{55mm}` paragraphs 55 mm wide

The `\operatorname` macro used in the example above is part of the `amsmath` package.

Including graphics

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

Encapsulated 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 $\text{\LaTeX}$ 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}
```

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}`

Figures and references

Larger diagrams interfere with page breaking. They are best placed into a figure environment, so that $\text{\LaTeX}$ 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!

Build tools for L^AT_EX

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 L^AT_EX documents where file modification timestamps indicate that this is necessary.

Graphics editor xfig

- ▶ Its *.fig files have a simple plain-text format that can be edited manually or generated by scripts, and that yields useful diffs.
- ▶ Can export *.eps or *.pdf files
- ▶ Can also produce figures in which $\text{\LaTeX}$ 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 $\text{\LaTeX}$ 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 $\text{\LaTeX}$ meta characters. Use `\input{*.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, pnmtops, Inkscape, MATLAB, R, gnuplot, Python+matplotlib

Bibliographic references

Academic writing: detailed references to prior work

- ▶ to support arguments
- ▶ to document familiarity with the field
- ▶ to help newcomers to the field
- ▶ to give due academic credit
- ▶ to join the bibliometric graph of the field (recommendation systems)
- ▶ to trigger notifications to cited authors (e.g. Google Scholar alerts)

A bibliographic reference needs to contain all information needed to help readers (and increasingly search engines!) to unambiguously identify and locate the text referred to. Minimum: First two authors, full title, date, indication of type of publication (usually implicitly via other details).

Inter-library loan required many details: names and locations of publishers or the volume, issue and page numbers in printed journals. Today, most readers retrieve scientific publications (with the exception of books) digitally, so online identifiers such as URIs or DOIs are now more important.

Many URLs (e.g., to authors' home pages, "grey literature") do not last long, but can still later be retrieved via archival services (e.g., archive.org). Therefore: accompany URLs with the date on which you accessed them. Try to HTTP redirect your own URLs when moving web space.

Bibliographic references – styles

Four widely used styles:

- ▶ **short-title** – most common in books in the humanities, e.g. “Hart, *Hart’s Rules*, p. 52”
- ▶ **author-date** – most common in science and social science, e.g. “Goossens (1997)” or “(Goossens, 1997b)”
- ▶ **author-number** – follows the name of the author(s) with a per-author number (rare with L^AT_EX), e.g. “Goossens (2)”
- ▶ **number-only** – all publications sequentially numbered in bibliography, made popular by L^AT_EX, now very widely used in computer science, e.g. “[67]”

Collect full references at the end of the document (bibliography), each chapter, or page of first mention (footnotes).

With number-only bibliography: for important work, include author(s) name and year in first mention in the text. Avoid abusing numerical references as nouns (bad: “as shown by [17]”, nice: “as Knuth showed in 1988 [17]”).

When referencing books, also include a chapter or page number.

Bibliographic references – manual approach

L^AT_EX can also manage numerical references to a bibliography.

Append near the end of your document (usually after conclusions and before appendices) your list of bibliographic items, assigning each an alphanumeric identifier:

```
\begin{thebibliography}{9}

\bibitem{Lamport94} Leslie Lamport: \LaTeX\ -- A document preparation
system : User's guide and reference manual. Addison-Wesley, 1994.

\bibitem{Companion} Frank Mittelbach, Ulrike Fischer: The \LaTeX\
Companion. 3rd edition, Addison-Wesley, 2023.

\bibitem{Dupr98} Lyn Dupr\ 'e: BUGS in Writing: A guide to debugging your
prose. 2nd edition, Addison-Wesley, 1998.

\end{thebibliography}
```

Then refer to these in the text using `\cite`:

Finer details on the use of en~dashes are explained
by Lamport~\cite{Lamport94} or Dupr\ 'e~\cite[page~134]{Dupr98}.

`\cite` can take an optional note in `[]`, e.g. for a page or section number.

`\bibitem` can take an optional label in `[]`.

`\begin{thebibliography}{99}` needs a “widest label” as an argument to plan indentation.

Automatic compilation of references with BibT_EX

Consider using BibT_EX if you

- ▶ expect to write several publications, sharing many references, each formatted to different house styles;
- ▶ want number-only references sorted in order of first reference;
- ▶ want automatic author-year references (use package natbib);
- ▶ want to use a reference database.

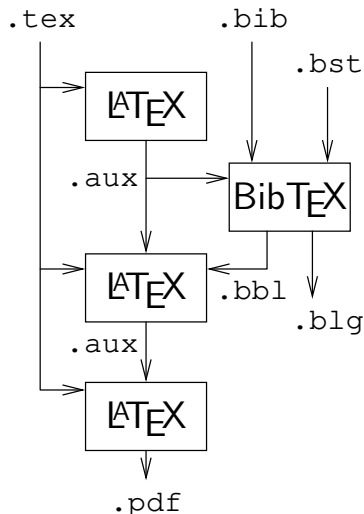
Provide .bib bibliographic database files and a .bst bibliographic style file (often provided by publisher). Load these as in

```
\bibliography{articles,group-refs}  
\bibliographystyle{unsrt}
```

Details:

Lamport, Section 4.3 and Appendix B.

L^AT_EX Companion, Chapter 12.



Example *.bib file entry

```
@TechReport{UCAM-CL-TR-123,  
  author =      {Herbert, John},  
  title =      {Case Study of the {Cambridge Fast Ring ECL}  
                Chip Using {HOL}},  
  year =      1988,  
  month =      feb,  
  institution = {University of Cambridge, Computer Laboratory},  
  number =      {UCAM-CL-TR-123}  
}
```

Important: Many U.S. publishers' house styles use "title case" for publication titles, where most words (other than articles and short prepositions) start with a capital letter.

This form of emphasis is not appropriate for bibliographic references and therefore many Bib_T_EX styles convert titles to lowercase.

Protect names (proper nouns) and acronyms in titles from automatic lowercase-ification by surrounding them with {...}.

Title case:	Case Study of the Cambridge Fast Ring ECL Chip Using HOL
Sentence case:	Case study of the Cambridge Fast Ring ECL chip using HOL
Bad-Bib _T _E X case:	Case study of the cambridge fast ring ecl chip using hol

Exercise 1: Copy file `example.tex` from slide 6, run “`pdflatex example`” twice (why?), and then “`okular example.pdf &`” (Linux) to see the output.

Exercise 2: Read pages 1–64 of the $\text{\LaTeX}$ book, then write your CV with $\text{\LaTeX}$, convert the result into PDF, and put it onto your homepage.

For information on how to set up homepages locally:

<https://www.srcf.net/>

Exercise 3: In a job interview for a position as a subeditor of a technical journal, your skills in spotting typographic mistakes made by $\text{\LaTeX}$ beginners are tested with this example text:

The -7 dB loss ($\pm 2\text{dB}$) shown on pp. 7-9 can be attributed to the $f(t) = \sin(2\pi ft)$ signal, where t is the the time and $f = 48\text{Khz}$ is the "sampling frequency".

Can you spot all 14 mistakes? Write down both the probable original incorrect $\text{\LaTeX}$ source text and a corrected version.

<https://www.cl.cam.ac.uk/~mgk25/e3>