

# The ShorTeX package

Trevor Campbell, Jonathan Huggins, and Jeff Negrea

Updated April 10, 2023

## Abstract

The purpose of the ShorTeX (meta)package is to make the process of typesetting typical mathematical documents in L<sup>A</sup>T<sub>E</sub>X more efficient, and the resulting code easier to read. It achieves this by (1) providing an extensive, internally consistent, and easy to learn set of macro shorthands, (2) incorporating a set of packages that are dedicated to reducing manual coding effort, and (3) incorporating a collection of very common / standard boilerplate packages.

## 1 Usage and package options

Include ShorTeX by adding `\usepackage{shortex}` to the preamble of your document. ShorTeX will include and configure many common packages for you (e.g., `graphicx`, `subcaption`, `hyperref`, `algorithm`, `algpseudocode`, `amsmath`, among others), so you do not need to explicitly include and set these up yourself. If you are writing a document that must use a specific style file (e.g., for a conference or journal) that itself includes some of these packages, we recommend editing those style files to remove the package imports.

The ShorTeX package has a few options:

**manualnumbering** Do not include `autonum.sty`. This disables automatic equation numbering.

**blackhypersetup** Switch hyperlinks, citations, references, etc. to be typeset in black font. The default is dark blue.

**You must compile your document 4 times when using ShorTeX** to ensure that equation numbers and references update properly.

## 2 Packages included in ShorTeX

### 2.1 Internal packages

The following packages provide features that are used internally in ShorTeX for macro definitions etc.

**xifthen**, **xstring**, **xspace**, **xargs** asdf

## 2.2 Typical packages

We include a set of typical packages to avoid the problem of having to remember precisely what order to import each one, which options, compatibility, etc

It is worth noting that we did not include a standard bibliography package in ShorTeX (e.g., `natbib.sty`). This is because...

```
mathrsfs,dsfont,amsmath,amssymb,amsthm,bm,bbm,amsfonts,mathtools,thmtools
asdf
```

```
hyperref asdf
```

```
color asdf
```

```
algorithm, algpseudocode asdf
```

```
graphicx asdf
```

## 2.3 Improvement packages

**cleveref** Typically to use a reference in L<sup>A</sup>T<sub>E</sub>X, you have to write the name of the type of reference yourself. For example, if you want to reference a figure, you would have to write something like:

```
In Figure \ref{fig:first}, you can see...
```

Or for multiple figures, you might use:

```
Figures \ref{fig:first}, \ref{fig:second},
and \ref{fig:third} show that...
```

The **cleveref** package simplifies this process significantly. Use the `\cref` command to automatically typeset the names of the objects you're referencing (including properly handling multiple references). The above two examples become

```
In \cref{fig:first}, you can see...
```

and

```
\cref{fig:first,fig:second,fig:third} show that...
```

This works for many different reference types (Figure, Algorithm, Equation, Table, etc), and can be extended if needed. See the **cleveref** documentation at <https://ctan.org/pkg/cleveref?lang=en> and the homepage at <https://www.dr-qubit.org/cleveref.html> for more information.

**autonum** Typically when you typeset equations, you have to choose between `$...$, $$...$$, \begin{align}...\end{align}, \begin{aligned}...\end{aligned}, \begin{equation}...\end{equation}`, not to mention starred versions of those environments and `\nonumber/\notag` commands, depending on whether/where you want equation numbers, display or in-text math, etc. This leads to verbose, inconsistent code.

The **autonum** package automatically decides which equations to provide numbers based on *which equations you reference*. So when using ShorTeX, you only need two commands for math mode: single dollar signs `$...$` for inline math, and **align** environments (redefined in ShorTeX to be `\[...\]`) for display math.<sup>1</sup>

For example, if you create the following display math,

```
\[
  a+b = c \label{eq:the_equation}
\]
```

then if you use the command `\cref{eq:the_equation}` somewhere in the document, that equation will automatically be assigned a number. If not, it won't get a number. See the **autonum** package documentation <https://ctan.org/pkg/autonum?lang=en> for more information.

**nicefrac** asdf

**crossreftools** asdf

**multirow** asdf

**wrapfig** asdf

**caption,subcaption** asdf

**microtype** asdf

**booktabs** asdf

**import,subfiles** asdf

**url** asdf

---

<sup>1</sup>Note that there are minor differences between **align** and the other display math environments like **equation**. But after 10+ years of using L<sup>A</sup>T<sub>E</sub>X, I have not ever encountered a case where it mattered much. That being said, ShorTeX does not *disable* any functionality, so you can use other environments anywhere you feel it is necessary.

## 3 Shortened commands and macros

### 3.1 Environments

Environment Name	Syntax
algorithm	<code>\balg...\ealg</code>
algorithmic	<code>\balgc...\ealgc</code>
table	<code>\btabs...\etabs</code>
subtable	<code>\bsubtab...\esubtab</code>
tabular	<code>\btabr...\etabr</code>
figure	<code>\bfig...\efig</code>
figure*	<code>\bfigs...\efigs</code>
subfigure	<code>\bsubfig...\esubfig</code>
centering	<code>\bcent...\ecent</code>
align	<code>\[...\]</code>
aligned	<code>\&lt;...\&gt;</code>
theorem	<code>\bthm...\ethm</code>
lemma	<code>\blem...\elem</code>
proposition	<code>\bprop...\eprop</code>
corollary	<code>\bcor...\ecor</code>
conjecture	<code>\bconj...\econj</code>
definition	<code>\bdef...\edef</code>
assumption	<code>\bassump...\eassump</code>
example	<code>\bexa...\eexa</code>
remark	<code>\brmk...\ermk</code>
fact	<code>\bfact...\efact</code>
exercise	<code>\bexer...\eexer</code>
proof	<code>\bprf...\eprf</code>
proof of	<code>\bprfof{\cref{proof_label}}...\eprfof</code>

### 3.2 Brackets and bracket-like functions

You can specify a bracket size using `*` for `\left` and `\right` or one of the standard size choices (`\big`, `\Big`, `\bigg`, `\Bigg`).

Description	Example	Text style	Display style
Round brackets (i.e., parentheses)	<code>\rbra{\frac{x}{y}}</code>	$(\frac{x}{y})$	$(\frac{x}{y})$
Curly brackets	<code>\cbra*{\frac{x}{y}}</code>	$\{\frac{x}{y}\}$	$\{\frac{x}{y}\}$
Square brackets	<code>\sbra[\bigg]{\frac{x}{y}}</code>	$[\frac{x}{y}]$	$[\frac{x}{y}]$

Many other bracket-like, semantic commands are also available:

Description	Example	Text style	Display style
Absolute value	<code>\abs{\frac{x}{y}}</code>	$ \frac{x}{y} $	$ \frac{x}{y} $
Set	<code>\set{\frac{x}{y}, \frac{y}{z}}</code>	$\{\frac{x}{y}, \frac{y}{z}\}$	$\{\frac{x}{y}, \frac{y}{z}\}$
Floor	<code>\floor{\frac{x}{y}}</code>	$\lfloor \frac{x}{y} \rfloor$	$\lfloor \frac{x}{y} \rfloor$
Ceiling	<code>\ceil{\frac{x}{y}}</code>	$\lceil \frac{x}{y} \rceil$	$\lceil \frac{x}{y} \rceil$
Norm	<code>\norm{\frac{x}{y}}</code>	$\ \frac{x}{y}\ $	$\ \frac{x}{y}\ $
Inner product	<code>\inner{\frac{x}{y}}{\frac{y}{z}}</code>	$\langle \frac{x}{y}, \frac{y}{z} \rangle$	$\langle \frac{x}{y}, \frac{y}{z} \rangle$
Cardinality	<code>\card{\wh A}</code>	$ \hat{A} $	$ \hat{A} $

The norm and inner product commands also have versions with a subscript argument:

Description	Example	Text style	Display style
Norm with subscript	<code>\normsub*{\frac{x}{y}}{2}</code>	$\ \frac{x}{y}\ _2$	$\ \frac{x}{y}\ _2$
Inner product with subscript	<code>\innersub*{\frac{x}{y}}{z}{2}</code>	$\langle \frac{x}{y}, z \rangle_2$	$\langle \frac{x}{y}, z \rangle_2$

### 3.3 $L_p$ Spaces and Operators

Description	Example	Text style	Display style
$L_p$ space	<code>\Lp{2}</code>	$L_2$	$L_2$
$L_p$ space for specified measure	<code>\Lpmeas{2}{\hmu}</code>	$L_2(\hat{\mu})$	$L_2(\hat{\mu})$
	<code>\Lpmeas[\Big]{2}{\hmu}</code>	$L_2\left(\hat{\mu}\right)$	$L_2\left(\hat{\mu}\right)$
$L_p$ norm	<code>\Lpnorm{\hGamma}{2}</code>	$\ \hat{\Gamma}\ _{L_2}$	$\ \hat{\Gamma}\ _{L_2}$
	<code>\Lpnorm*{\hGamma}{2}</code>	$\left\ \hat{\Gamma}\right\ _{L_2}$	$\left\ \hat{\Gamma}\right\ _{L_2}$
	<code>\Lpnorm*{\Gamma}{2}</code>	$\ \Gamma\ _{L_2}$	$\ \Gamma\ _{L_2}$
	<code>\left\Vert{\hGamma}\right\Vert_2</code>	$\left\ \hat{\Gamma}\right\ _2$	$\left\ \hat{\Gamma}\right\ _2$
	<code>\left\Vert{\Gamma}\right\Vert_2</code>	$\ \Gamma\ _2$	$\ \Gamma\ _2$
$L_p$ norm for specified measure	<code>\Lpmeasnorm{\hGamma}{2}{\hmu}</code>	$\ \hat{\Gamma}\ _{L_2(\hat{\mu})}$	$\ \hat{\Gamma}\ _{L_2(\hat{\mu})}$
	<code>\Lpmeasnorm[\Big]{\hGamma}{2}{\hmu}</code>	$\left\ \hat{\Gamma}\right\ _{L_2(\hat{\mu})}$	$\left\ \hat{\Gamma}\right\ _{L_2(\hat{\mu})}$
$L_p$ inner product	<code>\Lpinner{\hGamma}{\Gamma}{2}</code>	$\langle\hat{\Gamma},\Gamma\rangle_{L_2}$	$\langle\hat{\Gamma},\Gamma\rangle_{L_2}$
	<code>\Lpinner*{\hGamma}{\Gamma}{2}</code>	$\left\langle\hat{\Gamma},\Gamma\right\rangle_{L_2}$	$\left\langle\hat{\Gamma},\Gamma\right\rangle_{L_2}$
$L_p$ inner product for specified measure	<code>\Lpmeasinner{\hGamma}{\Gamma}{2}{\hmu}</code>	$\langle\hat{\Gamma},\Gamma\rangle_{L_2(\hat{\mu})}$	$\langle\hat{\Gamma},\Gamma\rangle_{L_2(\hat{\mu})}$
	<code>\Lpmeasinner[\big]{\hGamma}{\Gamma}{2}{\hmu}</code>	$\left\langle\hat{\Gamma},\Gamma\right\rangle_{L_2(\hat{\mu})}$	$\left\langle\hat{\Gamma},\Gamma\right\rangle_{L_2(\hat{\mu})}$

### 3.4 annotation commands

<code>\barA</code>	$\bar{A}$
<code>\bara</code>	$\bar{a}$
<code>\bA</code>	$\bar{A}$
<code>\bB</code>	$\bar{B}$
<code>\balpha</code>	$\bar{\alpha}$
<code>\bGamma</code>	$\bar{\Gamma}$
<code>\mcA</code>	$\mathcal{A}$
<code>\hmcA</code>	$\hat{\mathcal{A}}$
<code>\mfA</code>	$\mathfrak{A}$
<code>\mfa</code>	$\mathfrak{a}$
<code>\bmfa</code>	$\mathfrak{A}$
<code>\bmfa</code>	$\mathfrak{a}$
<code>\hA</code>	$\hat{A}$
<code>\ha</code>	$\hat{a}$
<code>\halpha</code>	$\hat{\alpha}$
<code>\hGamma</code>	$\hat{\Gamma}$
<code>\bhA</code>	$\hat{A}$
<code>\bha</code>	$\hat{a}$
<code>\bhalpha</code>	$\hat{\alpha}$
<code>\bhGamma</code>	$\hat{\Gamma}$
<code>\whA</code>	$\hat{A}$
<code>\wha</code>	$\hat{a}$
<code>\tdA</code>	$\tilde{A}$
<code>\tda</code>	$\tilde{a}$
<code>\tdalpha</code>	$\tilde{\alpha}$
<code>\tdGamma</code>	$\tilde{\Gamma}$
<code>\btdA</code>	$\tilde{A}$
<code>\btda</code>	$\tilde{a}$
<code>\btdalpha</code>	$\tilde{\alpha}$
<code>\btdGamma</code>	$\tilde{\Gamma}$
<code>\biA</code>	$\mathbf{A}$
<code>\bia</code>	$\mathbf{a}$
<code>\bhiA</code>	$\hat{\mathbf{A}}$
<code>\bhia</code>	$\hat{\mathbf{a}}$

### 3.5 new approach to annotation commands

Out of the box we can get all sorts of cool fonts in math mode using `\f[<fontcodestring>]A`. For the time being I only implemented 2 fonts and 1 accent. These can be expanded once everything else is set. We see the effect of single font codes:

`\f[b]A`     $\mathbf{A}$   
`\f[k]A`     $\mathbb{A}$   
`\f[h]A`     $\hat{A}$

And multiple font codes:

```

\fbhA  $\hat{\mathbf{A}}$ 
\fhbA  $\hat{\mathbf{A}}$ 
\fhkA  $\hat{\mathbf{A}}$ 
\fhkA  $\hat{\mathbf{A}}$ 

```

Note that these are expanded in the reverse of the order they appear: the font code furthest to the right is applied first. This matches the order that the corresponding commands would appear in TeX code.

We can also avoid typing [] for some combinations of font codes we sue frequently. To do this, use `\parsefontstylesstrings{<fcstr1>,<fcstr2>,...}<alphabet>` as demonstrated below. For “ease of use” we have implemented `\upperCaseRomanLetters` and `\lowerCaseRomanLetters`

```

\parsefontstylesstrings{{hb},{hk}}{ABCDEFGF} ...
\parsefontstylesstrings{{hb},{hk}}{\lowerCaseRomanLetters} ...
\fhbA  $\hat{\mathbf{A}}$ 
\fhkB  $\hat{\mathbf{B}}$ 
\fhbx  $\hat{\mathbf{x}}$ 
\fhby  $\hat{\mathbf{y}}$ 
\fhkz  $\hat{\mathbf{F}}$ 

```

Since `\mathbb<lowercaseletter>` is defined to give weird characters, our macros do the same.

## 4 Example Document

TODO: a full example in basic latex versus shortex