

The unimath-plain-X_YTeX package

Zhang Tingxuan

2022/11/23 Version 0.0y

Abstract

unimath-plain-X_YTeX package provides OpenType math font support in *plain* T_EX format. The unimath-plain-X_YTeX package needs X_YTeX.

1 How to use this package?

Please notice again that you're using *plain* format but not L^AT_EX format. If you are using L^AT_EX format, please use unicode-math package instead.

In your document, write

```
\input unimath-plain-xetex
```

Then compile your document with xetex, you can get OpenType math support in your document. The package will set the math font “Latin Modern Math” with “Latin Modern” text fonts in default. To change the font, you can define some names before loading the package. For example,

```
\def\mainfontname{TeX Gyre Termes}  
\def\sansfontname{TeX Gyre Heros}  
\def\monofontname{TeX Gyre Cursors}  
\def\mathfontname{TeX Gyre Termes Math}  
\input unimath-plain-xetex
```

Your text fonts will be set in the first 3 lines and your math font will be set in the fourth line.

Currently, the package supports only font family names to use, if you want to use the file names, you can revise the code in unimath-plain-xetex.tex.

2 Text font commands

The package provides text font commands in the format of

`\<pt-size>\<family>\<series>\<shape>`

such as `\tensfbfit`, `\twelvebf`, etc. The packages provides font commands in size of 5 pt, 7 pt, 9 pt, 10 pt, 12 pt and 20 pt, such as `\fiverm`, `\sevensf`, `\twelvett`, etc.

Take ten point as an example,

Serif		
	Upright	<i>Italic</i>
Medium	<code>\tenrm</code>	<code>\tenit</code>
Bold	<code>\tenbf</code>	<code>\tenbfit</code>
Sans		
	Upright	<i>Italic</i>
Medium	<code>\tensf</code>	<code>\tensfit</code>
Bold	<code>\tensfbf</code>	<code>\tensfbfit</code>
Typewriter		
	Upright	<i>Italic</i>
Medium	<code>\tentt</code>	<code>\tenttit</code>
Bold	<code>\tenttbfb</code>	<code>\tenttbfit</code>

The font commands can be used as those provided in `plain.tex`, for example, `{\tenbfit ABC}` yields ***ABC***.

But if you wouldn't like to remember that many commands, you can write

`\makefontcmdcompatible{\<pt-size>}`

after loading the package. For example, if you write

`\makefontcmdcompatible{ten}`

the `\<family>-\<series>-\<shape>` order of `\ten<some>` commands can be write randomly: writing `\tenbfsfit` is the same as `\tensfbfit`.

You can also get more text font commands through the `\genfontcmd` command:

```
\genfontcmd{<pt-size>}{<dimension>}
```

For example,

```
\genfontcmd{fortyfour}{44pt}
\genfontcmd{verytiny}{2bp}
```

will make commands like `\fortyfourrm` and `\verytinysfbfit` available.

3 Math font commands

You can input math formulae just like using traditional plain \TeX . But OpenType math font is loaded. For example, `\a{\bf0}=\{\bf0\}` yields $a\mathbf{0} = \mathbf{0}$. Available math font commands are listed below:

```
\rm, \bf, \it, \bfit, \sf, \sfbf, \sfit, \sfbfit,
\tt, \cal, \calbf, \bb, \bbit, \frak, \frakbf
```

The *<family>-<series>-<shape>* order of these commands allows being random; `cal` can be replaced by `scr`.

This package uses `unicode-math-symbols.tex` to generate math symbol commands, the source file can be found in `unicode-math` package. To find all of the math symbol commands, you can execute

```
texdoc unimath-symbols
```

in Terminal.

You can also input Unicode math characters in your document's source file. For example, `\int_a^b y\,dx` yields $\int_a^b y\,dx$ and `\int_a^b y\,dx.` yields

$$\int_a^b y\,dx.$$

4 Primes (')

The package changed “'” command in math mode, which yields prime(s) in superscript. After loading this package, the primes produced by “'” will be turned into Unicode primes:

<i>Input</i>	<i>yields</i>	<i>Unicode slot</i>
'	'	U+2032
''	''	U+2033
'''	'''	U+2034
''''	''''	U+2057
'''''	'''''	$(U+2032) \times 5$
\vdots	\vdots	\vdots
$(') \times N$	$(') \times N$	$(U+2032) \times N \quad (N > 4)$

However, some OpenType math fonts don't contain some of the characters above. For example, there is no U+2032, U+2033 or U+2034 in Erewhon Math. When using such fonts, you can “turn off” the Unicode primes easily by inputting the following line *after* loading the package:

```
\unicodeprimesoff
```

Then when you input “'” n times you will get n primes, and each prime's Unicode character slot is U+2032, even when $n \leq 4$.

If the primes contain more than one single prime encoded U+2032, a negative math skip will inserted between every two single primes. This math skip is defined as `\betweenprimeskip` and its default value is -2.7μ , which is proper for Latin Modern Math. To change the value of `\betweenprimeskip` is just like change any math skip, for example,

```
\betweenprimeskip=-1mu
```

This line should also written *after* loading the package.