

# The `xfp` package

## Floating Point Unit

The L<sup>A</sup>T<sub>E</sub>X3 Project\*

Released 2019-10-11

This package provides a L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> document-level interface to the L<sup>A</sup>T<sub>E</sub>X3 floating point unit (part of `expl3`). It also provides a parallel integer expression interface for convenience.

---

`\fpeval` ★

The expandable command `\fpeval` takes as its argument a floating point expression and produces a result using the normal rules of mathematics. As this command is expandable it can be used where T<sub>E</sub>X requires a number and for example within a low-level `\edef` operation to give a purely numerical result.

Briefly, the floating point expressions may comprise:

- Basic arithmetic: addition  $x + y$ , subtraction  $x - y$ , multiplication  $x * y$ , division  $x / y$ , square root  $\sqrt{x}$ , and parentheses.
- Comparison operators:  $x < y$ ,  $x <= y$ ,  $x >? y$ ,  $x != y$  *etc.*
- Boolean logic: sign `sign  $x$` , negation `! $x$` , conjunction  `$x$  &&  $y$` , disjunction  `$x$  ||  $y$` , ternary operator  `$x$ ?  $y$ :  $z$` .
- Exponentials: `exp  $x$` , `ln  $x$` ,  `$x$  $y$` .
- Integer factorial: `fact  $x$` .
- Trigonometry: `sin  $x$` , `cos  $x$` , `tan  $x$` , `cot  $x$` , `sec  $x$` , `csc  $x$`  expecting their arguments in radians, and `sind  $x$` , `cosd  $x$` , `tand  $x$` , `cotd  $x$` , `secd  $x$` , `cscd  $x$`  expecting their arguments in degrees.
- Inverse trigonometric functions: `asin  $x$` , `acos  $x$` , `atan  $x$` , `acot  $x$` , `asec  $x$` , `acsc  $x$`  giving a result in radians, and `asind  $x$` , `acosd  $x$` , `atand  $x$` , `acotd  $x$` , `asecd  $x$` , `acscd  $x$`  giving a result in degrees.
- Extrema: `max( $x_1, x_2, \dots$ )`, `min( $x_1, x_2, \dots$ )`, `abs( $x$ )`.
- Rounding functions, controlled by two optional values,  $n$  (number of places, 0 by default) and  $t$  (behavior on a tie, NaN by default):
  - `trunc( $x, n$ )` rounds towards zero,
  - `floor( $x, n$ )` rounds towards  $-\infty$ ,

---

\*E-mail: [latex-team@latex-project.org](mailto:latex-team@latex-project.org)

- `ceil(x, n)` rounds towards  $+\infty$ ,
- `round(x, n, t)` rounds to the closest value, with ties rounded to an even value by default, towards zero if  $t = 0$ , towards  $+\infty$  if  $t > 0$  and towards  $-\infty$  if  $t < 0$ .

- Random numbers: `rand()`, `randint(m, n)`.
- Constants: `pi`, `deg` (one degree in radians).
- Dimensions, automatically expressed in points, *e.g.*, `pc` is 12.
- Automatic conversion (no need for `\number`) of integer, dimension, and skip variables to floating points numbers, expressing dimensions in points and ignoring the stretch and shrink components of skips.
- Tuples:  $(x_1, \dots, x_n)$  that can be added together, multiplied or divided by a floating point number, and nested.

An example of use could be the following.

`\LaTeX{}` can now compute:  $\frac{\sin(3.5)}{2} + 2 \cdot 10^{-3}$   
`= \fpeval{sin(3.5)/2 + 2e-3}` \$.

---

`\inteval` \*

The expandable command `\inteval` takes as its argument an integer expression and produces a result using the normal rules of mathematics. The operations recognised are `+`, `-`, `*` and `/` plus parentheses. Division occurs with *rounding*, and ties are rounded away from zero. As this command is expandable it can be used where `TEX` requires a number and for example within a low-level `\edef` operation to give a purely numerical result.

An example of use could be the following.

`\LaTeX{}` can now compute: The sum of the numbers is  $\inteval{1 + 2 + 3}$  \$.

## Index

The italic numbers denote the pages where the corresponding entry is described, numbers underlined point to the definition, all others indicate the places where it is used.

<b>E</b>		<b>I</b>	
<code>\edef</code> .....	<i>1, 2</i>	<code>\inteval</code> .....	<i>2</i>
<b>F</b>		<b>N</b>	
<code>\fpeval</code> .....	<i>1</i>	<code>\number</code> .....	<i>2</i>