The `math` addon provides ways of writing commonly used mathemetical operations and symbols. It is accompanied by scripts for more specific uses, currently `math/func.ijs` and `math/sets.ijs`.

The `array` addon is loaded by `math`, and is used freely here. If you are unfamiliar with the array functions used here, see the array tutorial.

1 Greek

The `greek` addon defines nouns for all the greek letters. It is used here for convenience.

```
tau = +:pi
```

$$\tau = 2\pi$$

```
mathlist Gamma, Delta, Theta
```

Γ, Δ, and Θ

2 Mathematical functions

The `of` verb is the same as math's function application. It can be applied to a list of arguments as well. `of` always parenthesizes the argument.

```
f of x
```

$$f(x)$$

```
f of r,theta,phi
```

$$f(r, \theta, \phi)$$

```
(Gamma of 3r2) = -:%:pi
```

$$\Gamma\left(\frac{3}{2}\right) = \frac{\sqrt{\pi}}{2}$$

With large arguments functions use `\left` and `\right` for their delimiters, which results in poor spacing without further configuration. To avoid this, add the following lines (also found in `../tutorial.sty`) to the preamble of your document:

```
\let\originalleft\left
\let\originalright\right
\renewcommand{\left}{%
    \mathopen{}\mathclose\bgroup\originalleft}
\renewcommand{\right}{\aftergroup\egroup\originalright}
```

3 Math operation types

Three new types of operations are defined by `math`:

Style	Example		Declaration
Sum	`sum`	$\sum_j a_j$	`DeclareSumOp`
Math	`sin`	$\sin\theta$	`DeclareMathOp`
Limit	`lim`	$\lim_{n\to\infty} a_n$	`DeclareMathLimOp`

In each case, nearly all operations built into LaTeX are remapped. If you need one that is not or have defined your own operation, simply use the appropriate declaration verb from the right column.

3.1 Sum-style operations

Functions like the big-sigma sum and integrals take a bound as a left argument and the term to sum (integrate, etc.) as the right argument. The bound has length up to two: the first element is the lower bound and the second is the upper bound. It may also be omitted entirely.

```
(6%~*:pi) =~ (\_infty ,~ n=1) sum %*:n
```

$$\sum_{n=1}^{\infty} \frac{1}{n^2} = \frac{\pi^2}{6}$$

```
X = (alpha in A) bigcup U_ alpha
```

$$X = \bigcup_{\alpha\in A} U_\alpha$$

```
(int@:(*&dx) = +&C) ^x
```

$$\int e^x dx = e^x + C$$

3.2 Math operations

Operations like `sin`, `exp`, and so on are defined as verbs. For these functions, the argument is parenthesized only if it is sufficiently complicated.

```
list sin"0 (+: , ] , -:) theta
```

$$\sin(2\theta), \sin\theta, \sin\left(\frac{\theta}{2}\right)$$

```
='+/ dim"0 V ([ , ker@] , of~) T
```

$$\dim V = \dim(\ker T) + \dim(T(V))$$

```
(gcd = \+./) 42 98 126
```

$$\gcd(42, 98, 126) = 14$$

Note that `log` is not defined—use `^.` instead!

3.3 Limit-style operations

Limit operations are monadic or take a single left argument, which is placed as a subscript.

```
min ((-~^) st in&RR) x
```

$$\min\{\, e^x - x \mid x \in \mathbb{R} \,\}$$

```
(n to _) lim | a %/@:\_ (,<:)n
```

$$\lim_{n\to\infty} \left| \frac{a_n}{a_{n-1}} \right|$$

4 Miscellaneous

The operations `oplus` and `otimes` are declared in `math`:

```
b ((a otimes [.@oplus) = oplus&[.&(a&otimes)) c
```

$$a \otimes (b \oplus c) = (a \otimes b) \oplus (a \otimes c)$$

5 Sets

The `math/sets` addon provides some standard set nouns as well as verbs for common ways to write sets.

The following sets are predefined:

`NN`	$\mathbb{N}$
`ZZ`	$\mathbb{Z}$
`QQ`	$\mathbb{Q}$
`RR`	$\mathbb{R}$
`CC`	$\mathbb{C}$
`II`	$[0, 1]$

The `DeclareSet` verb is the declaration for a set, which is written in blackboard bold.

```
HH DeclareSet H
HH in~ list \1,i,j,k
```

$$1, i, j, k \in \mathbb{H}$$

To place a list of elements in a set, use the `set` verb. To invoke "set-builder notation," use `st`. The left argument to `st` is the variable and the right argument is the predicate.

```
set \(/:~ , */&:>/ (^ i.@>:)&.>/ __ q: 12)
```

$$\{1, 2, 3, 4, 6, 12\}$$

```
(x in RR) st (*:x) < 2
```

$$\{\, x \in \mathbb{R} \,|\, x^2 < 2 \,\}$$

The verbs `subset`, `supset`, `in`, and `setminus` are defined as math operations.

```
iff DeclareOp 'Longleftrightarrow'
A (subset iff supset~ iff \_varnothing =~ setminus) B
```

$$A \subset B \iff B \supset A \iff A \setminus B = \varnothing$$

The infix operations `and`, `quot`, `by`, and `comma` are defined.

```
(x and y) in S
```

$$x, y \in S$$

```
(quot p&*) ZZ
```

$$\mathbb{Z}/p\mathbb{Z}$$

```
int (^x) by dx  NB. space with \,
```

$$\int e^x \, dx$$

```
(a=3) comma (b=5)  NB. space with \quad
```

$$a = 3, \quad b = 5$$

6 Functions

The following functions are defined by `math/func`.

The operations `from` (as `colon`), `to` (as `rightarrow`), and `mapsto` are declared:

```
(f from RR to RR)  comma  x mapsto *:x
```

$$f \colon \mathbb{R} \to \mathbb{R}, \quad x \mapsto x^2$$

Function composition is `o`.

```
((f o g)&of = f of g&of) x
```

$$f \circ g(x) = f(g(x))$$

The inverse is `inv`, and you can apply a function's inverse with `if` (that is, `of` with the first character changed). The function `res` gives the restriction of a function to a domain, using a vertical bar.

```
a (* * *&inv) b
```

$$aba^{-1}b^{-1}$$

```
([.f res A) if x
```

$$(f|_A)^{-1}(x)$$

The verbs `star` and `prime` do the obvious:

```
(star H_1) of S^2
```

$$H_1^*(S^2)$$

```
(prime^:2 f) of x
```

$$f''(x)$$

The conjunction `u commutes v` gives `u@:v = v&:u`.

```
*: commutes % x
```

$$\left(\frac{1}{x}\right)^2 = \frac{1}{x^2}$$

```
z + commutes \\cdot w
```

$$\overline{z \cdot w} = \overline{z} \cdot \overline{w}$$