op_arith — Arithmetic operators

Description Syntax Remarks and examples Conformability Diagnostics Also see

Description

The above operators perform basic arithmetic.

Syntax

$a + b$	addition
$a - b$	subtraction
$a * b$	multiplication
a / b	division
$a \hat{} b$	power
$-a$	negation

where a and b may be numeric scalars, vectors, or matrices.

Remarks and examples

Also see [M-2] **op_colon** for the `:+`, `:-`, `:*`, and `:/` operators. Colon operators have relaxed conformability restrictions.

The `*` and `:*` multiplication operators can also perform string duplication—`3*"a" = "aaa"`—see [M-5] **strdup()**.

Conformability

$a + b$, $a - b$:

a:	$r \times c$	
b:	$r \times c$	
result:	$r \times c$	

$a * b$:

a:	$k \times n$	$k \times n$	1×1
b:	$n \times m$	1×1	$n \times m$
result:	$k \times m$	$k \times n$	$n \times m$

a / b:

a:	$r \times c$
b:	1×1
result:	$r \times c$

$a \hat{} b$:

a:	1×1
b:	1×1
result:	1×1

−*a*:

a:	$r \times c$
result:	$r \times c$

Diagnostics

All operators return missing when arguments are missing.

a`*`*b* with *a*: $k \times 0$ and *b*: $0 \times m$ returns a $k \times m$ matrix of zeros.

a`/`*b* returns missing when *b*`==0` or when *a*`/`*b* would result in overflow.

a`^`*b* returns a real when both *a* and *b* are real; thus, `(-4)^.5` evaluates to missing, whereas `(-4+0i)^.5` evaluates to `2i`.

a`^`*b* returns missing on overflow.

Also see

[M-2] **exp** — Expressions

[M-2] **Intro** — Language definition