The Special Functions Reference Manual

Special functions in Common Lisp, version 1.1

Steve Nunez <steve@symbolics.tech>

Table of Contents

1 Systems

The main system appears first, followed by any subsystem dependency.

1.1 special-functions

Author Steve Nunez <steve@symbolics.tech>

License MS-PL

Description
Special functions in Common Lisp

Version 1.1

Dependencies
- num-utils
- float-features

Source [special-functions.asd], page 3, (file)

Directory s:/src/special-functions/

Components
- [pkgdcl.lisp], page 3, (file)
- [utils.lisp], page 3, (file)
- [erf.lisp], page 3, (file)
- [gamma.lisp], page 3, (file)
- [log-gamma.lisp], page 4, (file)
- [factorial.lisp], page 4, (file)

2 Files

Files are sorted by type and then listed depth-first from the systems components trees.

2.1 Lisp

2.1.1 special-functions.asd

Location /src/special-functions/special-functions.asd

Systems [special-functions], page 1, (system)

2.1.2 special-functions/pkgdcl.lisp

Parent [special-functions], page 1, (system)

Location pkgdcl.lisp

Packages [special-functions], page 5,

2.1.3 special-functions/utils.lisp

Dependency
[pkgdcl.lisp], page 3, (file)

Parent [special-functions], page 1, (system)

Location utils.lisp

Internal Definitions

- [+square-root-2-pi+], page 8, (constant)
- [decode-float64], page 8, (function)
- [encode-float64], page 8, (function)
- [sin-pi], page 9, (function)

2.1.4 special-functions/erf.lisp

Dependency
[utils.lisp], page 3, (file)

Parent [special-functions], page 1, (system)

Location erf.lisp

Exported Definitions

- [erf], page 7, (function)
- [erfc], page 7, (function)
- [inverse-erf], page 7, (function)
- [inverse-erfc], page 7, (function)

Internal Definitions
[inverse-error], page 9, (function)

2.1.5 special-functions/gamma.lisp

Dependency
[erf.lisp], page 3, (file)

Parent [special-functions], page 1, (system)

Location `gamma.lisp`

Exported Definitions

[`gamma`], page 7, (function)

Internal Definitions

- [`gamma-inverse-small`], page 8, (function)
- [`gamma-medium`], page 8, (function)
- [`maxgamd`], page 8, (constant)
- [`sign-gamma`], page 9, (function)
- [`stirling`], page 9, (function)
- [`tiny`], page 8, (constant)

2.1.6 `special-functions/log-gamma.lisp`

Dependency

[`gamma.lisp`], page 3, (file)

Parent [`special-functions`], page 1, (system)

Location `log-gamma.lisp`

Exported Definitions

[`log-gamma`], page 7, (function)

2.1.7 `special-functions/factorial.lisp`

Dependency

[`log-gamma.lisp`], page 4, (file)

Parent [`special-functions`], page 1, (system)

Location `factorial.lisp`

Exported Definitions

[`factorial`], page 7, (function)

Internal Definitions

- [`factorial-table`], page 8, (special variable)
- [`ramanujan`], page 9, (function)
- [`sam-ramanujan`], page 9, (function)

3 Packages

Packages are listed by definition order.

3.1 special-functions

Source [pkgdcl.lisp], page 3, (file)

Nicknames

- specfun
- spfn

Use List

- num-utils.arithmetic
- num-utils.polynomial
- common-lisp

Exported Definitions

- [erf], page 7, (function)
- [erfc], page 7, (function)
- [factorial], page 7, (function)
- [gamma], page 7, (function)
- [inverse-erf], page 7, (function)
- [inverse-erfc], page 7, (function)
- [log-gamma], page 7, (function)

Internal Definitions

- [+square-root-2-pi+], page 8, (constant)
- [decode-float64], page 8, (function)
- [encode-float64], page 8, (function)
- [factorial-table], page 8, (special variable)
- [gamma-inverse-small], page 8, (function)
- [gamma-medium], page 8, (function)
- [inverse-error], page 9, (function)
- [maxgamd], page 8, (constant)
- [ramanujan], page 9, (function)
- [sam-ramanujan], page 9, (function)
- [sign-gamma], page 9, (function)
- [sin-pi], page 9, (function)
- [stirling], page 9, (function)
- [tiny], page 8, (constant)

4 Definitions

Definitions are sorted by export status, category, package, and then by lexicographic order.

4.1 Exported definitions

4.1.1 Functions

`erf` *N* [Function]
Returns the error function of n

Package [special-functions], page 5,

Source [erf.lisp], page 3, (file)

`erfc` *X* [Function]
Returns the complementary error function of x

Package [special-functions], page 5,

Source [erf.lisp], page 3, (file)

`factorial` *X* [Function]
Return the factorial value X! for X <= MAX-FACTORIAL; DOUBLE-FLOAT-POSITIVE-INFINITY if x < 0. X must be an INTEGER.

Package [special-functions], page 5,

Source [factorial.lisp], page 4, (file)

`gamma` *X* [Function]
Return gamma(x), x <= MAXGAMD; NAN/RTE if x is a non-positive integer

Package [special-functions], page 5,

Source [gamma.lisp], page 3, (file)

`inverse-erf` *X* [Function]
Return the inverse function of erf: (erf (inverse-erf x)) = x, -1 < x < 1

Package [special-functions], page 5,

Source [erf.lisp], page 3, (file)

`inverse-erfc` *X* [Function]
Return the inverse function of erfc: (erfc (inverse-erfc x)) = x, 0 < x < 2

Package [special-functions], page 5,

Source [erf.lisp], page 3, (file)

`log-gamma` *N* [Function]
Return the logarithm of gamma(x)

Package [special-functions], page 5,

Source [log-gamma.lisp], page 4, (file)

4.2 Internal definitions

4.2.1 Constants

`+square-root-2-pi+` [Constant]

Package [special-functions], page 5,

Source [utils.lisp], page 3, (file)

`maxgamd` [Constant]

Maximum argument for gamma

Package [special-functions], page 5,

Source [gamma.lisp], page 3, (file)

`tiny` [Constant]

Package [special-functions], page 5,

Source [gamma.lisp], page 3, (file)

4.2.2 Special variables

`factorial-table` [Special Variable]

Table of factorials for integer values up to 100

Package [special-functions], page 5,

Source [factorial.lisp], page 4, (file)

4.2.3 Functions

`decode-float64` *X* [Function]

Convert the (unsigned-byte 64) bit representation into a native double-float

Package [special-functions], page 5,

Source [utils.lisp], page 3, (file)

`encode-float64` *X* [Function]

Returns the bit representation of the double-float X as an (unsigned-byte 64)

Package [special-functions], page 5,

Source [utils.lisp], page 3, (file)

`gamma-inverse-small` *X* [Function]

Return 1/gamma(x) for `|x|` < 0.03125

Package [special-functions], page 5,

Source [gamma.lisp], page 3, (file)

`gamma-medium` *X* [Function]

Return gamma(x), `|x|` <= 13, x negative integer produces div by 0

Package [special-functions], page 5,

Source [gamma.lisp], page 3, (file)

`inverse-error` P Q [Function]
Return value of inverse error function: erf_inv(p) if p <= 0.5, erfc_inv(q) otherwise

Package [`special-functions`], page 5,

Source [`erf.lisp`], page 3, (file)

`ramanujan` X [Function]
Ramanujan's original approximation of n!

Package [`special-functions`], page 5,

Source [`factorial.lisp`], page 4, (file)

`sam-ramanujan` X [Function]
Modification of Ramanujan's approximation of n! by Sidney A. Morris

Package [`special-functions`], page 5,

Source [`factorial.lisp`], page 4, (file)

`sign-gamma` X [Function]
Return sign(gamma(x)), invalid for 0 or negative integer

Package [`special-functions`], page 5,

Source [`gamma.lisp`], page 3, (file)

`sin-pi` X [Function]
Returns (sin (* pi x))

Package [`special-functions`], page 5,

Source [`utils.lisp`], page 3, (file)

`stirling` X [Function]
Return (gamma x) for x > 13

Package [`special-functions`], page 5,

Source [`gamma.lisp`], page 3, (file)

Appendix A Indexes

A.1 Concepts

F

L

S

A.2 Functions

A.3 Variables

A.4 Data types

P

S