

Package ‘RcppFastFloat’

August 21, 2021

Type Package

Title 'Rcpp' Bindings for the 'fast_float' Header-Only Library for
Number Parsing

Version 0.0.3

Date 2021-08-21

Author Dirk Eddelbuettel, Brendan Knapp

Maintainer Dirk Eddelbuettel <edd@debian.org>

Description Converting ascii text into (floating-point) numeric values is a very common problem. The 'fast_float' header-only C++ library by Daniel Lemire does it very well and very fast at up to or over to 1 gigabyte per second as described in more detail in <[arXiv:2101.11408](https://arxiv.org/abs/2101.11408)>. 'fast_float' is licensed under the Apache 2.0 license and provided here for use by other R packages via a simple 'LinkingTo:' statement.

License GPL (>= 2)

Imports Rcpp

LinkingTo Rcpp

Suggests tinytest

URL <https://github.com/eddelbuettel/rcppfastfloat/>,
<https://dirk.eddelbuettel.com/code/rcpp.fastfloat.html>

BugReports <https://github.com/eddelbuettel/rcppfastfloat/issues>

RoxygenNote 6.0.1

Encoding UTF-8

NeedsCompilation yes

Repository CRAN

Date/Publication 2021-08-21 13:20:06 UTC

R topics documented:

as.double2	2
parseExample	3

as.double2

Ultra efficient string-to-double Conversion

Description

For character vectors, `as.double2()` is a drop-in replacement for `base::as.double()`.

Usage

```
as.double2(x)
```

Arguments

`x` A vector of type character.

See Also

`as.double()`

Examples

```
set.seed(8675309)
input <- sample(c(
  paste0(" \r\n\t\f\v", c(0.0, sqrt(seq(1, 10))), " \r\n\t\f\v"),
  c("NaN", "-NaN", "nan", "-nan",
    "Inf", "-Inf", "inf", "-inf", "infinity", "-infinity",
    NA_character_,
    " 1970-01-01", "1970-01-02 ")
))
input

suppressWarnings(as.double2(input)) # NAs introduced by coercion

comparison <- suppressWarnings(
  matrix(c(as.double(input), as.double2(input)),
        ncol = 2L,
        dimnames = list(NULL, c("as.double()", "as.double2()")))
)
comparison

all.equal(comparison[, "as.double()"], comparison[, "as.double2()"])
```

`parseExample`*Floating Point Parsing Example*

Description

This example is adapted from the example of the upstream README.md file, and generalized to be called from R with variable input.

Usage

```
parseExample(input = "3.1416 xyz ", verbose = TRUE)
```

Arguments

<code>input</code>	A character variable with text to parse including a simple default
<code>verbose</code>	A boolean variable to show or suppress progress, defaults to true

Value

A floating point scalar is returned on success; in case of parsing failure the function exists via `stop()`.

Examples

```
parseExample()
```

Index

`as.double2`, 2

`parseExample`, 3