Package ‘natcpp’

July 13, 2021

Type Package

Title Fast C++ Primitives for the 'NeuroAnatomy Toolbox'

Version 0.1.0

Description Fast functions implemented in C++ via 'Rcpp' to support the 'NeuroAnatomy Toolbox' ('nat') ecosystem. These functions provide large speed-ups for basic manipulation of neuronal skeletons over pure R functions found in the 'nat' package. The expectation is that end users will not use this package directly, but instead the 'nat' package will automatically use routines from this package when it is available to enable large performance gains.

License GPL (>= 3)

URL https://github.com/natverse/natcpp

BugReports https://github.com/natverse/natcpp/issues

Imports Rcpp (>= 1.0.6)

Suggests spelling, testthat (>= 3.0.0)

LinkingTo Rcpp

Config/testthat/edition 3

Encoding UTF-8

Language en-GB

RoxygenNote 7.1.1

NeedsCompilation yes

Author Gregory Jefferis [aut, cre] (<https://orcid.org/0000-0002-0587-9355>)

Maintainer Gregory Jefferis <jefferis@gmail.com>

Repository CRAN

Date/Publication 2021-07-13 14:50:02 UTC
R topics documented:

- \texttt{c_EdgeListFromSegList} ......................................................... 2
- \texttt{c_listlengths} ................................................................. 3
- \texttt{c_seglengths} ................................................................. 3
- \texttt{c_topntail} ................................................................. 4

Index

\begin{verbatim}
c_EdgeListFromSegList  Turn a segment list into an edgelist suitable for constructing an ngraph
\end{verbatim}

Description

Turn a segment list into an edgelist suitable for constructing an ngraph

Usage

\begin{verbatim}
c_EdgeListFromSegList(L)
\end{verbatim}

Arguments

\begin{verbatim}
L  a list containing integer vectors from \texttt{as.seglist}
\end{verbatim}

Details

It is up to the caller to generate the \texttt{seglist}. Note that isolated points will be dropped since they have no edges.

Value

An integer matrix of \(N\) rows and 2 columns

Examples

\begin{verbatim}
## Not run:
library(nat)
# make a neuron with multiple subtrees
n=prune_vertices(Cell07PNs[[1]], 48L)
# Must use flatten=T if including all subtrees
sl=as.seglist(n, all = TRUE, flatten = TRUE)
c_EdgeListFromSegList(sl)
## End(Not run)
\end{verbatim}
**c_listlengths**

A simple function to compute the lengths of the elements of an R list

**Description**

A simple function to compute the lengths of the elements of an R list

**Usage**

```r
c_listlengths(L)
```

**Arguments**

- `L`: a list

**Details**

This is equivalent to the `base::lengths` however it it much faster for long lists (and somewhat slower for short ones).

**Value**

An integer vector containing the length of each element of `L`

---

**c_seglengths**

Compute summed segment lengths or total cable

**Description**

c_seglengths computes the summed segment length equivalent to `nat::seglengths(sumsegment = T)`
c_total_cable computes the summed total cable for a whole neuron. Its intended use is the `nat::summary.neuron` function.

**Usage**

```r
c_seglengths(sl, x, y, z)
c_total_cable(sl, x, y, z)
```

**Arguments**

- `sl`: A seglist with 1-indices into vectors `x,y,z`
- `x`, `y`, `z`: Numeric vectors with 3D coordinate data (which could be columns from a data frame)
c_topntail

Find the first and last elements of all vectors in a list

Description

c_topntail returns an 2xN matrix containing the start and end of each of the vectors in the input list. Length 0 vectors are ignored, while length 1 vectors are duplicated.

For c_topntail_list, a list of the same length as L having the same elements when their length is <=2 or the first and last elements when length>2.

Usage

c_topntail(L)

c_topntail_list(L)

Arguments

L a list containing integer vectors, typically a seglist

Value

For c_topntail an integer matrix. For c_topntail_list a list.
Index

c_EdgeListFromSegList, 2
 c_listlengths, 3
 c_seglengths, 3
 c_topntail, 4
 c_topntail_list (c_topntail), 4
 c_total_cable (c_seglengths), 3