# Package 'RcppInt64'

April 30, 2024

**Type** Package

**Title** 'Rcpp'-Based Helper Functions to Pass 'Int64' and 'nanotime' Values Between 'R' and 'C++'

**Version** 0.0.5

**Date** 2024-04-30

**Description** 'Int64' values can be created and accessed via the 'bit64' package and its 'integer64' class which package the 'int64' representation cleverly into a 'double'. The 'nanotime' packages builds on this to support nanosecond-resolution timestamps. This packages helps conversions between 'R' and 'C++' via several helper functions provided via a single header file. A complete example client package is included as an illustration.

**URL** <https://github.com/eddelbuettel/rcppint64>

**BugReports** <https://github.com/eddelbuettel/rcppint64/issues>

**License** GPL (>= 2)

**Imports** Rcpp (>= 1.0.8)

**LinkingTo** Rcpp

**Suggests** tinytest, bit64, nanotime

**RoxygenNote** 6.0.1

**Encoding** UTF-8

**NeedsCompilation** yes

**Author** Dirk Eddelbuettel [aut, cre] (<<https://orcid.org/0000-0001-6419-907X>>)

**Maintainer** Dirk Eddelbuettel <edd@debian.org>

**Repository** CRAN

**Date/Publication** 2024-04-30 12:22:36 UTC

## R topics documented:

---

RcppInt64-package          *'Rcpp'-Based Helper Functions to Pass 'Int64' and 'nanotime' Values Between 'R' and 'C++'*

---

### Description

'Int64' values can be created and accessed via the 'bit64' package and its 'integer64' class which package the 'int64' representation cleverly into a 'double'. The 'nanotime' packages builds on this to support nanosecond-resolution timestamps. This packages helps conversions between 'R' and 'C++' via several helper functions provided via a single header file. A complete example client package is included as an illustration.

### Package Content

Index of help topics:

```
Int64toInt64            Integer64 to Integer64 round-trip demo
NanotimeToNanotime      nanotime to nanotime round-trip demo
RcppInt64-package       'Rcpp'-Based Helper Functions to Pass 'Int64'
                        and 'nanotime' Values Between 'R' and 'C++'
```

### Maintainer

Dirk Eddelbuettel <edd@debian.org>

### Author(s)

Dirk Eddelbuettel [aut, cre] (<https://orcid.org/0000-0001-6419-907X>)

---

Int64toInt64              *Integer64 to Integer64 round-trip demo*

---

### Description

This function takes an integer64-valued input vector, converts it to the equivalent int64_t vector in C++, displays each element after first adding one, and returns the modified vector.

### Usage

```
Int64toInt64(vec)
```

### Arguments

vec             An integer64-classed vector from R

## Value

A modified `integer64` vector where each element increased by one

## Examples

```
# generate all powers of 10 fro 0 .. 18
if (requireNamespace("bit64", quietly=TRUE)) {
    v <- bit64::as.integer64(10^seq(0,18))
    # pass them to function which will add one to each, print  and return
    Int64toInt64(v)
}
```

---

NanotimeToNanotime      *nanotime to nanotime round-trip demo*

---

## Description

This function takes an nanotime-valued input vector, converts it to the equivalent `int64_t` vector in C++, displays each element after first adding one, and returns the modified vector.

## Usage

```
NanotimeToNanotime(vec)
```

## Arguments

vec             A nanotime-classed vector from R

## Value

A modified `nanotime` vector where each element increased by one

## Examples

```
# generate all powers of 10 fro 0 .. 18
if (requireNamespace("nanotime", quietly=TRUE)) {
    v <- nanotime::as.nanotime(10^seq(0,18))
    # pass them to function which will add one to each, print  and return
    NanotimeToNanotime(v)
}
```

# Index