

Package ‘fletcher’

July 15, 2026

Type Package

Title Compute Fletcher Checksums (16, 32, and 64 Bit)

Version 0.1.0

Description Computes Fletcher's position-dependent checksum in 16-, 32-, and 64-bit widths. Fletcher's checksum, devised by John G. Fletcher, provides error-detection properties approaching a cyclic redundancy check at lower computational cost. Input is processed as little-endian words with the final partial word zero-padded, so results are reproducible across platforms.

License MIT + file LICENSE

Encoding UTF-8

URL <https://github.com/chrislongros/fletcher>

BugReports <https://github.com/chrislongros/fletcher/issues>

Suggests tinytest

NeedsCompilation yes

RoxygenNote 7.3.2

Author Christos Longros [aut, cre]

Maintainer Christos Longros <chris.longros@gmail.com>

Repository CRAN

Date/Publication 2026-07-15 17:50:13 UTC

Contents

fletcher	2
Index	3

fletcher	<i>Compute a Fletcher checksum</i>
----------	------------------------------------

Description

Computes Fletcher's position-dependent checksum. Input is processed as little-endian words with the final partial word zero-padded, so the result is reproducible across platforms.

Usage

```
fletcher(x, width = 32L)
```

Arguments

x	A character string, a raw vector, or an integer/numeric vector of byte values (0-255) coercible with <code>as.raw</code> . Character input is converted with <code>charToRaw</code> .
width	Checksum width in bits; one of 16, 32, or 64.

Value

A lowercase hexadecimal string of width / 4 characters.

Examples

```
fletcher("abcde", 16) # "c8f0"  
fletcher("abcde", 32) # "f04fc729"  
fletcher("abcde", 64) # "c8c6c527646362c6"
```

Index

fletcher, [2](#)