

# Package: bencoding (via r-universe)

September 17, 2024

**Version** 0.1-1

**Title** Tools for handing Bencoding

**Author** Simon Urbanek <Simon.Urbaneck@r-project.org>

**Maintainer** Simon Urbanek <Simon.Urbaneck@r-project.org>

**Depends** R (>= 2.9.0)

**Description** This package provides tools for handling Bencoding.

**License** GPL-2 | GPL-3

**URL** <http://www.rforge.net/bencoding>

**Repository** <https://s-u.r-universe.dev>

**RemoteUrl** <https://github.com/s-u/bencoding>

**RemoteRef** HEAD

**RemoteSha** df095280ab522609f46a2a3dbb663983a505ceea

## Contents

readBenc .....	1
<b>Index</b>	<b>3</b>

---

readBenc	<i>Read content encoded in Bencoding</i>
----------	--

---

## Description

readBenc reads content in Bencoding into R objects.

## Usage

readBenc(what)

**Arguments**

what                    character string of file name(s), binary connection or a raw vector

**Details**

Bencoding defines integers, strings, lists and dictionaries which are stored as integer (or real) scalars, string scalars, pairlists and named pairlists respectively.

**Value**

Decoded content. If what is a character vector of more than one element then the result is a list equivalent to `lapply(what, readBenc)`.

**Note**

In principle Bencoding supports integers of arbitrary precision, but R only supports signed integers up to 32-bit. All values outside that range will be stored as (double-precision) reals. This means that only signed integers up to 53-bit precision will be represented without loss of precision.

R does not allow strings with embedded NULs, so any string with a value less than TAB is returned as a raw vector.

**Author(s)**

Simon Urbanek

**Examples**

```
ex <- charToRaw("d1:ai100e1:b13:fooi123456789012345e3:baree")
readBenc(ex)
```

# Index

\* **manip**  
  readBenc, [1](#)

readBenc, [1](#)