

# Package ‘crbcc’

August 4, 2026

**Title** C R Bytecode Compiler

**Version** 0.1.0

**Description** Implements a bytecode compiler for 'R' in 'C', targeting functional parity with the base 'compiler' package, while focusing on better performance. The architecture mostly mirrors the original 'GNU-R' compiler as described in <https://homepage.cs.uiowa.edu/~luke/R/compiler/compiler.pdf>.

**License** GPL (>= 2)

**Encoding** UTF-8

**RoxygenNote** 7.1.1

**Depends** R (>= 4.5.0)

**Suggests** compiler

**URL** <https://github.com/PRL-PRG/crbcc>

**BugReports** <https://github.com/PRL-PRG/crbcc/issues>

**NeedsCompilation** yes

**Author** Josef Malý [aut, cre],  
Filip Křikava [ths]

**Maintainer** Josef Malý <malyjos4@fit.cvut.cz>

**Repository** CRAN

**Date/Publication** 2026-08-04 09:40:20 UTC

## Contents

|         |   |
|---------|---|
| cmpfile | 2 |
| cmpfun  | 3 |
| compile | 3 |
| loadcmp | 4 |

|       |   |
|-------|---|
| Index | 5 |
|-------|---|

---

`cmpfile`*Compile an R Source File*

---

## Description

Compiles all top-level expressions in an R source file to bytecode and saves the result to an output file.

## Usage

```
cmpfile(infile, outfile, ascii = FALSE, env = .GlobalEnv,  
        verbose = FALSE, options = NULL, version = NULL)
```

## Arguments

|                      |                                                                                                                    |
|----------------------|--------------------------------------------------------------------------------------------------------------------|
| <code>infile</code>  | Path to the input R source file.                                                                                   |
| <code>outfile</code> | Path to the output file. If missing, defaults to the input file name with extension replaced by <code>.Rc</code> . |
| <code>ascii</code>   | Logical; whether to write an ASCII file.                                                                           |
| <code>env</code>     | The top-level environment in which to evaluate expressions.                                                        |
| <code>verbose</code> | Logical; Not yet implemented, print compilation progress messages.                                                 |
| <code>options</code> | List of compilation options.                                                                                       |
| <code>version</code> | Optional file format version.                                                                                      |

## Value

Invisible NULL. Output is written to file as a side effect.

## Examples

```
tmp_r <- tempfile(fileext = ".R")  
tmp_rc <- tempfile(fileext = ".Rc")  
writeLines("x <- 1; x + 1", tmp_r)  
cmpfile(tmp_r, tmp_rc)
```

---

`cmpfun`*Compile an R Function to Bytecode*

---

**Description**

Compiles an R closure to bytecode using the C-based compiler backend.

**Usage**

```
cmpfun(fun, options = list())
```

**Arguments**

|                      |                                      |
|----------------------|--------------------------------------|
| <code>fun</code>     | A closure to compile to bytecode.    |
| <code>options</code> | A named list of compilation options. |

**Value**

A bytecode-compiled function.

**Examples**

```
f <- function(x) x + 1
cf <- cmpfun(f)
cf(10)
```

---

`compile`*Compile a Generic Language Object*

---

**Description**

Compiles a generic R language object (such as an expression or function) to bytecode using the C-based compiler backend.

**Usage**

```
compile(e, env = .GlobalEnv, options = NULL, srcref = NULL)
```

**Arguments**

|                      |                                                       |
|----------------------|-------------------------------------------------------|
| <code>e</code>       | An R language object to compile.                      |
| <code>env</code>     | The environment in which to evaluate the object.      |
| <code>options</code> | A list of compilation options, or <code>NULL</code> . |
| <code>srcref</code>  | Optional source reference information.                |

**Value**

The compiled object, or invisible NULL if compilation is only for side effect.

**Examples**

```
expr <- quote({ x <- 1; x + 2 })
compile(expr)
```

---

loadcmp

*Load a Compiled R File*


---

**Description**

Loads bytecode compiled by [cmpfile](#) into an environment.

**Usage**

```
loadcmp(file, envir)
```

**Arguments**

|       |                                                                                           |
|-------|-------------------------------------------------------------------------------------------|
| file  | Path to the compiled .Rc file.                                                            |
| envir | The environment to load into, use .GlobalEnv to load the code into current global session |

**Value**

Invisible NULL.

**Examples**

```
tmp_r <- tempfile(fileext = ".R")
tmp_rc <- tempfile(fileext = ".Rc")
writeLines("f <- function(x) x * 2", tmp_r)
cmpfile(tmp_r, tmp_rc)
loadcmp(tmp_rc, .GlobalEnv)
f(3)
```

# Index

cmpfile, [2](#), [4](#)

cmpfun, [3](#)

compile, [3](#)

loadcmp, [4](#)