

Package ‘giacR’

August 17, 2023

Title Interface to the Computer Algebra System 'Giac'

Version 1.0.0

Description 'Giac'

<https://www-fourier.ujf-grenoble.fr/~parisse/giac/doc/en/cascmd_en/cascmd_en.html>
is a general purpose symbolic algebra software. It powers the graphical interface 'Xcas'. This package allows to execute 'Giac' commands in 'R'.

License GPL-3

URL <https://github.com/stla/giacR>

BugReports <https://github.com/stla/giacR/issues>

Imports chromote (>= 0.1.2), jsonlite, pingr, processx, R6, utils

Encoding UTF-8

RoxygenNote 7.2.3

SystemRequirements Chromium-based browser (Google Chrome, Brave, ...)

NeedsCompilation no

Author Stéphane Laurent [aut, cre],
Renée De Graeve [cph] (Giac),
Bernard Parisse [cph] (Giac)

Maintainer Stéphane Laurent <laurent_step@outlook.fr>

Repository CRAN

Date/Publication 2023-08-17 06:52:39 UTC

R topics documented:

Giac	2
Index	5

Giac

R6 class to access to Giac

Description

Creates an object allowing to execute Giac commands.

Methods

Public methods:

- [Giac\\$new\(\)](#)
- [Giac\\$execute\(\)](#)
- [Giac\\$implicitization\(\)](#)
- [Giac\\$close\(\)](#)

Method new(): Create a new Giac instance.

Usage:

```
Giac$new(chromePath = find_chrome())
```

Arguments:

chromePath path to the Chrome executable (or Chromium, Brave, etc); if find_chrome() does not work, you can set the environment variable CHROMOTE_CHROME to the path and it will work

Returns: A Giac object.

Method execute(): Execute a Giac command.

Usage:

```
Giac$execute(command, timeout = 10000)
```

Arguments:

command the command to be executed given as a character string
timeout timeout in milliseconds

Returns: The result of the command in a character string.

Examples:

```
\donttest{if(!is.null(chromote::find_chrome())) {  
  giac <- Giac$new()  
  giac$execute("2 + 3/7")  
  giac$execute("integrate(ln(x))")  
  giac$close()  
}}
```

Method implicitization(): Gröbner implicitization (see examples)

Usage:

```
Giac$implicitization(
  equations,
  relations = "",
  variables,
  constants = "",
  timeout = 10000
)
```

Arguments:

equations comma-separated equations

relations comma-separated relations, or an empty string if there is no relation; the relations between the constants must be placed first, followed by the relations between the variables

variables comma-separated variables

constants comma-separated constants, or an empty string if there is no constant

timeout timeout in milliseconds

Returns: The implicitization of the equations.

Examples:

```
library(giacR)
\donttest{if(!is.null(chromote::find_chrome())) {
  giac <- Giac$new()
  giac$implicitization(
    equations = "x = a*cos(t), y = b*sin(t)",
    relations = "cos(t)^2 + sin(t)^2 = 1",
    variables = "cos(t), sin(t)",
    constants = "a, b"
  )
  giac$close()
}}
```

Method close(): Close a Giac session

Usage:

```
Giac$close()
```

Returns: TRUE or FALSE, whether the session has been closed.

Examples

```
## -----
## Method `Giac$execute`
## -----

if(!is.null(chromote::find_chrome())) {
  giac <- Giac$new()
  giac$execute("2 + 3/7")
  giac$execute("integrate(ln(x))")
  giac$close()
}
```

```
## -----  
## Method `Giac$implicitization`  
## -----  
  
library(giacR)  
if(!is.null(chromote::find_chrome())) {  
  giac <- Giac$new()  
  giac$implicitization(  
    equations = "x = a*cos(t), y = b*sin(t)",  
    relations = "cos(t)^2 + sin(t)^2 = 1",  
    variables = "cos(t), sin(t)",  
    constants = "a, b"  
  )  
  giac$close()  
}
```

Index

Giac, [2](#)