

# Package ‘Julia’

July 21, 2025

**Type** Package

**Title** Fractal Image Data Generator

**Version** 1.3.5

**Date** 2023-08-27

**Maintainer** Mehmet Suzen <mehmet.suzen@physics.org>

**URL** <https://github.com/msuzen/Julia>

**BugReports** <https://github.com/msuzen/Julia/issues>

**Description** Generates image data for fractals (Julia and Mandelbrot sets) on the complex plane in the given region and resolution. Benoit B Mandelbrot (1982).

**License** GPL-3

**NeedsCompilation** no

**Author** Mehmet Suzen [aut, cre]

**Repository** CRAN

**Date/Publication** 2023-08-27 16:20:05 UTC

## Contents

|                         |          |
|-------------------------|----------|
| JuliaImage . . . . .    | 2        |
| JuliaIterate . . . . .  | 3        |
| MandelImage . . . . .   | 4        |
| MandelIterate . . . . . | 5        |
| <b>Index</b>            | <b>7</b> |

JuliaImage

*Julia Set Generator in a Square Region*

---

**Description**

'JuliaImage' returns two dimensional array representing escape values from on the square region in complex plane. Escape values (which measures the number of iteration before the lenght of the complex value reaches to 2).

**Usage**

```
JuliaImage(imageN, centre, L, C)
```

**Arguments**

|        |                                                                              |
|--------|------------------------------------------------------------------------------|
| imageN | Number of pixels to equally space division of one side if the square region. |
| centre | A complex number that determines the centre of the square region             |
| L      | A side length of the square region on the complex plane.                     |
| C      | Complex coefficient                                                          |

**Details**

Julia Set is defined as the set of initial complex values where the  $z = z^2 + C$  does not diverge to infinity. C is an arbitrary complex constant that does not change during the iteration by definition.

**Value**

It returns a 2D array of real values from 0 to 1. The array correspods to image on the complex plane.

**Note**

Post processing to plot/color mapping of the Julia set for visualisation can be done by using the array generated. See examples to get a png output.

**Author(s)**

Mehmet Suzen <mehmet.suzen@physics.org>

**References**

Gaston Julia (1918) "Memoire sur l'iteration des fonctions rationnelles," Journal de Mathematiques Pures et Appliquees, vol. 8, pages 47-245.

**See Also**

[MandelImage](#)

**Examples**

```
#
# Generating png of the Julia set
# C is 1 minus the golden ratio
#
imageN <- 5; # increase this to see images
centre <- 0.0
L <- 4.0
C <- 1i-1.6180339887;# Golden Ratio
image <- JuliaImage(imageN,centre,L,C);
#library(png)
#file <- "julia1.png"
#writePNG(image,file); # possible visulation
#
# Generating png of the Julia set
# different coefficient.
#
imageN <- 5; # increase this to see images
centre <- 0.0
L <- 4.0
C <- -0.70176-0.3842i
image <- JuliaImage(imageN,centre,L,C);
#library(png)
#file <- "julia2.png"
#writePNG(image,file); # possible visulation
```

---

JuliaIterate

*JuliaIterate*


---

**Description**

'JuliaIterate' returns the number of iteration until a complex value diverges for the Julia map for a give complex number.

**Usage**

```
JuliaIterate(z, C)
```

**Arguments**

|   |                                                   |
|---|---------------------------------------------------|
| z | A complex coordinate (initial value for the map). |
| C | A complex constant.                               |

**Details**

'JuliaIterate' returns the number of iteration until a complex value diverges for the Julia map for a give complex number.

**Value**

Number of iterations.

**Note**

Iterative function.

**Author(s)**

Mehmet Suzen <mehmet.suzen@physics.org>

**References**

The Fractal Geometry of Nature, Benoit B. Mandelbrot, W.H.Freeman & Co Ltd (18 Nov 1982)

**See Also**

[JuliaIterate](#) and [MandelIterate](#)

**Examples**

```
z<-0+0i
C <- 1-1.6180339887;# Golden Ratio
it<- JuliaIterate(z,C)
```

---

MandelImage

*Mandelbrot Set Generator in a Square Domain*


---

**Description**

'MandelImage' returns two dimensional array representing escape values from on the square region in complex plane. Escape values (which measures the number of iteration before the length of the complex value reaches to 2.)

**Usage**

```
MandelImage(imageN, centre, L)
```

**Arguments**

|        |                                                                              |
|--------|------------------------------------------------------------------------------|
| imageN | Number of pixels to equally space division of one side if the square region. |
| centre | A complex number that determines the centre of the square region.            |
| L      | A side length of the square region on the complex plane.                     |

**Details**

Mandelbrot set is defined as the set of initial complex values where the  $z = z^2 + z_0$  does not diverge to infinity. Initial value for the map is taken to be zero and  $z_0$  is the complex coordinate.

**Value**

Returns a matrix.

**Note**

Returns a matrix

**Author(s)**

Mehmet Suzen <mehmet.suzen@physics.org>

**References**

The Fractal Geometry of Nature, Benoit B. Mandelbrot, W.H.Freeman & Co Ltd (18 Nov 1982)

**See Also**

[JuliaImage](#)

**Examples**

```
# png image
imageN <- 5; # increase this to see image
centre <- 0.0
L <- 4.0
image<-MandelImage(imageN,centre,L);
#file <- "mandelbrot1.png"
# writePNG(image,file); # possible visualisation
# Closer lookup to set
imageN <- 5;
centre <- -0.5
L <- 2.0
image<-MandelImage(imageN,centre,L);
# file <- "mandelbrot.png"
#writePNG(image,file); # possible visualisation
```

---

MandelIterate

*MandelIterate*

---

**Description**

'MandelIterate' returns the number of iteration until a complex value diverges for the Mandelbrot map for a give complex number.

**Usage**

MandelIterate(z\_0)

**Arguments**

`z_0`                    A complex coordinate (constant coefficient value for the map)

**Details**

Iterate function.

**Value**

Returns an integer

**Note**

Iterate function

**Author(s)**

Mehmet Suzen <mehmet.suzen@physics.org>

**References**

The Fractal Geometry of Nature, Benoit B. Mandelbrot, W.H.Freeman & Co Ltd (18 Nov 1982)

**See Also**

[JuliaIterate](#) and [MandelIterate](#)

**Examples**

```
z_0 <- 0-0.5i
it <- MandelIterate(z_0)
```

# Index

JuliaImage, [2](#), [5](#)

JuliaIterate, [3](#), [4](#), [6](#)

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

MandelIterate, [4](#), [5](#), [6](#)