# Package 'Brobdingnag'

August 13, 2018

Type Package

Title Very Large Numbers in R				
Version 1.2-6				
<b>Date</b> 2018-08-08				
Author Robin K. S. Hankin				
<b>Depends</b> R ( $\geq$ 2.13.0), methods				
Maintainer Robin K. S. Hankin <a href="mailto:hankin.robin@gmail.com">hankin.robin@gmail.com</a>				
<b>Description</b> Handles very large numbers in R. Real numbers are held using their natural logarithms, plus a logical flag indicating sign. The package includes a vignette that gives a step-by-step introduction to using S4 methods.				
LazyLoad yes				
License GPL				
Repository CRAN				
URL https://github.com/RobinHankin/Brobdingnag.git NeedsCompilation no Date/Publication 2018-08-13 13:20:03 UTC				
R topics documented:				
Brobdingnag-package				
Arith-methods				
as.numeric				
brob				
brob-class				
Compare-methods				
Complex				
Extract.brob				
getP				
glub				

Index		19
	swift-class	18
	sum	
	Print	
	plot	
	Math	15
	Logic	15
	length-methods	14
	glub-class	13

Very Large Numbers in R

Description

Handles very large numbers in R. Real numbers are held using their natural logarithms, plus a logical flag indicating sign. The package includes a vignette that gives a step-by-step introduction to using S4 methods.

#### **Details**

#### The DESCRIPTION file:

Brobdingnag-package

Package: Brobdingnag Type: Package

Title: Very Large Numbers in R

Version: 1.2-6 Date: 2018-08-08

Author: Robin K. S. Hankin Depends: R (>= 2.13.0), methods

Maintainer: Robin K. S. Hankin <a href="mailto:knakin.robin@gmail.com">hankin.robin@gmail.com</a>

Description: Handles very large numbers in R. Real numbers are held using their natural logarithms, plus a logical flag indic

LazyLoad: yes License: GPL Repository: CRAN

URL: https://github.com/RobinHankin/Brobdingnag.git

# Index of help topics:

Arith-methods Methods for Function Arith in package

Brobdingnag

Brobdingnag-package Very Large Numbers in R

Compare-methods Methods for Function Compare in Package

Brobdingnag

Re Real and imaginary manipulation

[.brob Extract or Replace Parts of brobs or glubs

Brobdingnag-package 3

abs Various logarithmic and circular functions for brobs as.numeric Coerces to numeric or complex form brob Brobdingnagian numbers brob-class Class "brob" cbrob Combine Brobdingnagian vectors getP Get and set methods for brob objects Glubbdubdribian numbers: complex numbers with glub Brobdingnagian real and imaginary parts glub-class Class "glub" length Get lengths of brobs and glubs Logical operations on brobs logic.brob Basic plotting of Brobs plot

print.brob Methods for printing brobs and glubs

sum Various summary statistics for brobs and glubs

swift-class Class "swift"

Real numbers are represented by two objects: a real, holding the logarithm of their absolute values; and a logical, indicating the sign. Multiplication and exponentiation are easy: the challenge is addition. This is achieved using the (trivial) identity  $\log(e^x + e^y) = x + \log(1 + e^{y-x})$  where, WLOG, y < x.

Complex numbers are stored as a pair of brobs: objects of class glub.

The package is a simple example of S4 methods.

However, it *could* be viewed as a cautionary tale: the underlying R concepts are easy yet the S4 implementation is long and difficult. I would not recommend using S4 methods for a package as simple as this; S3 methods would have been perfectly adequate. I would suggest that S4 methods should only be used when S3 methods are *demonstrably* inadequate.

#### Author(s)

Robin K. S. Hankin

Maintainer: Robin K. S. Hankin <a href="mailto:rhankin.robin@gmail.com">hankin.robin@gmail.com</a>

#### References

R. K. S. Hankin 2007. "Very Large Numbers in R: Introducing Package Brobdingnag". R News, volume 7, number 3, pages 15-16

```
googol <- as.brob(10)^100
googol
googol + googol/2
1/(googol + 1e99)
(1:10)^googol</pre>
```

4 Arith-methods

```
googolplex <- 10^googol
googolplex
googolplex * googol # practically the same as googolplex (!)</pre>
```

Arith-methods

Methods for Function Arith in package Brobdingnag

# Description

Methods for Arithmetic functions in package Brobdingnag: +, -, \*, /, ^

#### Note

The unary arithmetic functions (viz "+" and "-") do no coercion.

The binary arithmetic functions coerce numeric <op> brob to brob; and numeric <op> glub, complex <op> brob, and brob <op> glub, to glub.

## Author(s)

Robin K. S. Hankin

```
x <- as.brob(1:10)
y <- 1e10
x+y
as.numeric((x+y)-1e10)
x^(1/y)</pre>
```

as.numeric 5

as.numeric

Coerces to numeric or complex form

## **Description**

Coerces an object of class brob to numeric, or an object of class glub to complex

# Arguments

x Object of class brob or glub

... Further arguments (currently ignored)

## **Details**

Function as.numeric() coerces a brob to numeric; if given a glub, the imaginary component is ignored (and a warning given).

Function as.complex() coerces to complex.

## Note

If |x| is greater than .Machine\$double.xmax, then as.numeric(x) returns Inf or -Inf but no warning is given.

## Author(s)

Robin K. S. Hankin

```
a <- as.brob(1:10)
a <- cbrob(a, as.brob(10)^1e26)
a
as.numeric(a)
as.complex(10i + a)</pre>
```

6 brob

brob

Brobdingnagian numbers

## Description

Create, coerce to or test for a Brobdingnagian object

#### Usage

```
brob(x = double(), positive)
as.brob(x)
is.brob(x)
```

#### **Arguments**

x Quantity to be tested, coerced in to Brobdingnagian form

positive In function brob(), logical indicating whether the number is positive (actually,

positive or zero)

#### **Details**

Function as.brob() is the user's workhorse: use this to coerce numeric vectors to brobs.

Function is.brob() tests for its arguments being of class brob.

Function brob() takes argument x and returns a brob formally equal to  $e^x$ ; set argument positive to FALSE to return  $-e^x$ . Thus calling function  $\exp(x)$  simply returns  $\operatorname{brob}(x)$ . This function is not really intended for the end user: it is confusing and includes no argument checking. In general numerical work, use function as.brob() instead, although be aware that if you really really want  $e^{10^7}$ , you should use  $\operatorname{brob}(1e7)$ ; this would be an **exact** representation.

#### Note

Real numbers are represented by two objects: a real, holding the logarithm of their absolute values; and a logical, indicating the sign. Multiplication and exponentiation are easy: the challenge is addition. This is achieved using the (trivial) identity  $\log(e^x + e^y) = x + \log(1 + e^{y-x})$  where, WLOG, y < x.

Complex numbers are stored as a pair of brobs: objects of class glub.

The package is a simple example of S4 methods. However, it *could* be viewed as a cautionary tale: the underlying R concepts are easy yet the S4 implementation is long and difficult. I would not recommend using S4 methods for a package as simple as this; S3 methods would have been perfectly adequate. I would suggest that S4 methods should only be used when S3 methods are *demonstrably* inadequate.

The package has poor handling of NA and NaN. Currently, as.brob(1) + as.brob(c(1,NA)) returns an error.

brob-class 7

#### Author(s)

Robin K. S, Hankin

#### See Also

glub

#### **Examples**

```
googol <- as.brob(10)^100
googolplex <- 10^googol

(googolplex/googol) / googolplex
# Thus googolplex/googol == googolplex (!)

# use cbrob() instead of c() when Brobdingnagian numbers are involved: cbrob(4,exp(as.brob(1e55)))</pre>
```

brob-class

Class "brob"

## **Description**

The formal S4 class for Brobdingnagian numbers

# **Objects from the Class**

Objects *can* be created by calls of the form new("brob", ...) but this is not encouraged. Use functions brob() and, especially, as.brob() instead.

#### **Slots**

x: Object of class "numeric" holding the log of the absolute value of the number to be represented positive: Object of class "logical" indicating whether the number is positive (see Note, below)

#### **Extends**

```
Class "swift", directly.
```

## Note

Slot positive indicates non-negativity, as zero is conventionally considered to be "positive".

8 cbrob

#### Author(s)

Robin K. S. Hankin

#### See Also

```
glub-class,swift-class
```

#### **Examples**

```
new("brob",x=5,positive=TRUE) # not intended for the user
as.brob(5) # Standard user-oriented idiom
```

cbrob

Combine Brobdingnagian vectors

#### **Description**

Combine Brobdingnagian or Glubdubbdribian vectors through concatenation

#### Usage

```
cbrob(x, ...)
```

#### **Arguments**

x Brobdingnagian vector

... Other arguments coerced to brob form

#### **Details**

If any argument has class glub, all arguments are coerced to glubs. Otherwise, if any argument has class brob, all arguments are coerced to brobs.

Function cbrob() operates recursively, calling .cPair() repeatedly. Function .cPair() uses S4 method dispatch to call either .Brob.cpair() or .Glub.cpair() according to the classes of the arguments.

#### Note

As of R-2.4.0, it is apparently not possible to use S4 methods to redefine c() to coerce to class brob form and concatenate as expected. This would seem to be a reasonable interpretation of c() from the user's perspective.

Conceptually, the operation is simple: concatenate the value slot and the positive slot separately, then call brob() on the two resulting vectors. When concatenating glub objects, the real and imaginary components (being brobs) are concatenated using .Brob.cpair()

The choice of name—cbrob()—is not entirely logical. Because it operates consistently on brob and glub objects, it might be argued that cSwift() would be a more appropriate name.

Compare-methods 9

#### Author(s)

Robin K. S. Hankin; original idea due to John Chambers

# **Examples**

```
a <- as.brob(2)^1e-40
cbrob(1:4,4:1,a)
cbrob(1:4,a,1i)</pre>
```

Compare-methods

Methods for Function Compare in Package Brobdingnag

## **Description**

Methods for comparision (greater than, etc) in package Brobdingnag

#### Note

As for min() and max(), comparison is not entirely straightforward in the presence of NAs.

The low-level workhorses are .Brob.equal() for equality and .Brob.greater() for 'strictly greater than'. All other comparisons are calculated by combining these two.

Comparison [function .Brob.compare()] explicitly tests for a zero length argument and if given one returns logical(0) to match base behaviour.

# **Examples**

```
a <- as.brob(10)^(0.5 + 97:103)
a < 1e100
```

Complex

Real and imaginary manipulation

# Description

Get or set real and imaginary components of brobs or glubs.

10 Extract.brob

#### Usage

```
## S4 method for signature 'glub'
Re(z)
## S4 method for signature 'glub'
Im(z)
## S4 method for signature 'glub'
Mod(z)
## S4 method for signature 'glub'
Conj(z)
## S4 method for signature 'glub'
Arg(z)
Re(z) <- value
Im(z) <- value</pre>
```

#### **Arguments**

```
z object of class glub (or, in the case of Im<-() or Im(z) <- value, class brob) value object of class numeric or brob
```

#### Value

Functions Re() and Im() return an object of class brob; functions Re<-() and Im<-() return an object of class glub

## Author(s)

Robin K. S. Hankin

## **Examples**

```
a <- cbrob(1:10,brob(1e100))
Im(a) <- 11:1
```

Extract.brob

Extract or Replace Parts of brobs or glubs

## **Description**

Methods for "[" and "[<-", i.e., extraction or subsetting of brobs and glubs.

# Arguments

x Object of class brob or glubi elements to extract or replacevalue replacement value

getP 11

## Value

Always returns an object of the same class as x.

#### Note

If x is a numeric vector and y a brob, one might expect typing  $x[1] \leftarrow y$  to result in x being a brob. This is impossible, according to John Chambers.

## Author(s)

Robin K. S. Hankin

## **Examples**

```
a <- as.brob(10)^c(-100,0,100,1000,1e32)
a[4]
a[4] <- 1e100</pre>
```

getP

Get and set methods for brob objects

## **Description**

Get and set methods for brobs: sign and value

## Usage

```
getP(x)
getX(x)
sign(x) <- value</pre>
```

## **Arguments**

x Brobdingnagian objectvalue In function sign<-(), Boolean specifying whether the brob object is positive</li>

## Author(s)

Robin K. S. Hankin

# See Also

brob

12 glub

#### **Examples**

```
x <- as.brob(-10:10)
sign(x) <- TRUE
```

glub

Glubbdubdribian numbers: complex numbers with Brobdingnagian real and imaginary parts

#### **Description**

Create, coerce to or test for a Glubbdubdribian object

#### Usage

```
glub(real = double(), imag = double())
as.glub(x)
is.glub(x)
```

#### Arguments

real, imag Real and imaginary components of complex number: must be Brobdingnagian

numbers

x object to be coerced to or tested for Glubbdubdribian form

#### **Details**

Function glub() takes two arguments that are coerced to Brobdingnagian numbers and returns a complex number. This function is not really intended for the end user: it is confusing and includes no argument checking. Use function as.glub() instead.

Function as.glub() is the user's workhorse: use this to coerce numeric or complex vectors to Glubbdubdribian form.

Function is.glub() tests for its arguments being Glubbdubdribian.

#### Note

Function glub() uses recycling inherited from cbind().

#### Author(s)

Robin K. S. Hankin

#### See Also

brob

glub-class 13

#### **Examples**

glub-class

Class "glub"

#### **Description**

Complex Brobdingnagian numbers

#### **Objects from the Class**

A glub object holds two slots, both brobs, representing the real and imaginary components of a complex vector.

#### Slots

```
real: Object of class "brob" representing the real component imag: Object of class "brob" representing the imaginary component
```

#### **Extends**

```
Class "swift", directly.
```

#### Methods

```
.cPair signature(x = "brob", y = "glub"): ...
.cPair signature(x = "ANY", y = "glub"): ...
.cPair signature(x = "glub", y = "glub"): ...
.cPair signature(x = "glub", y = "ANY"): ...
.cPair signature(x = "glub", y = "brob"): ...
Im<- signature(x = "glub"): ...
Re<- signature(x = "glub"): ...</pre>
```

#### Author(s)

Robin K. S. Hankin

14 length-methods

## See Also

```
brob-class,swift-class
```

# **Examples**

```
a <- as.brob(45)
new("glub",real=a, imag=a)
as.brob(5+5i)  # standard colloquial R idiom</pre>
```

length-methods

Get lengths of brobs and glubs

# Description

Get lengths of brob and glub vectors

# Usage

```
## S4 method for signature 'brob'
length(x)
## S4 method for signature 'glub'
length(x)
```

## **Arguments**

Х

vector of class brob or glub

# Author(s)

Robin K. S. Hankin

```
x <- as.brob(-10:10)
length(x)</pre>
```

Logic 15

Logic

Logical operations on brobs

## **Description**

Logical operations on brobs are not supported

#### Note

The S4 group generic "Logic" appeared in R-2.4.0-patched.

Carrying out logical operations in this group will call .Brob.logic(), which reports an error.

Negation, "!", is not part of this group: attempting to negate a brob will not activate .Brob.logic(); an "invalid argument type" error is given instead.

#### Author(s)

Robin K. S. Hankin

#### **Examples**

```
## Not run:
!brob(10)
## End(Not run)
```

Math

Various logarithmic and circular functions for brobs

## **Description**

Various elementary functions for brobs

## **Arguments**

x Object of class brob (or sometimes glub)
base In function log(), the base of the logarithm

## **Details**

For brobs: apart from abs(), log(), exp(), sinh() and cosh(), these functions return f(as.numeric(x)) so are numeric; the exceptional functions return brobs.

For glubs: mostly direct transliteration of the appropriate formula; one might note that log(z) is defined as glub(log(Mod(x)), Arg(x)).

Print Print

#### Author(s)

Robin K. S. Hankin

## **Examples**

```
exp(as.brob(3000)) #exp(3000) is represented with zero error
```

plot

Basic plotting of Brobs

# Description

Plotting methods. Essentially, any brob is coerced to a numeric and any glub is coerced to a complex, and the argument or arguments are passed to plot().

## Usage

```
plot(x, y, ...)
```

## **Arguments**

x,y Brob or glub
... Further arguments passed to plot()

## Author(s)

Robin K. S. Hankin

## **Examples**

```
plot(as.brob(1:10))
```

Print

Methods for printing brobs and glubs

## **Description**

Methods for printing brobs and glubs nicely using exponential notation

## Usage

```
## $3 method for class 'brob'
print(x, ...)
## $3 method for class 'glub'
print(x, ...)
```

sum 17

#### **Arguments**

x An object of class brob or glub... Further arguments (currently ignored)

#### Author(s)

Robin K. S. Hankin

#### **Examples**

```
a <- as.brob(1:5)
dput(a)
a</pre>
```

sum

Various summary statistics for brobs and glubs

#### **Description**

Various summary statistics for brobs and glubs

#### **Arguments**

x,... Objects of class brob or, in the case of sum() and prob(), class glub
na.rm Boolean, with default FALSE meaning to interpret NAs literally and TRUE meaning

to ignore any such elements

#### **Details**

For a brob object, being NA is not entirely straightforward. The S4 method for is.na is too "strict" for some of the functions considered here. Consider max(a) where a includes only positive, fully specified, elements, and elements with known negative sign and exponents that include NA values. Here, max(a) is unambiguously determined.

Similar logic applies to min() and, by extension, range().

# Note

Function prod() is *very* slow for long glub vectors. It has to compute four Brobdingnagian products and two Brobdingnagian sums per element of its argument, and this takes a long time.

#### Author(s)

Robin K. S. Hankin

#### See Also

is.na

swift-class

# **Examples**

```
a <- as.brob(1:10)
max(cbrob(1:10,brob(NA,FALSE)))</pre>
```

swift-class

Class "swift"

# Description

A (virtual) class that extends brob and glub objects

# **Objects from the Class**

A virtual Class: No objects may be created from it.

# Methods

No methods defined with class "swift" in the signature.

# Author(s)

Robin K. S. Hankin

## See Also

brob-class,glub-class

# **Index**

Arith-methods, 4 as.numeric, 5 brob, 6 cbrob, 8 Compare-methods, 9 Complex, 9 Extract.brob, 10 getP, 11 glub, 12 length-methods, 14 Logic, 15 Math, 15 plot, 16 Print, 16 sum, 17 *Topic methods Arith-methods, 4 Compare-methods, 9 length-methods, 14 *Topic package Brobdingnag-package, 2 .cPair, ANY, ANY-method (brob-class), 7 .cPair, ANY, glub-method (brob-class), 7 .cPair, Prob, brob-method (brob-class), 7 .cPair, brob, complex-method	*Topic <b>classes</b>	[,brob-method(Extract.brob), 10
**Topic math  Arith-methods, 4     as. numeric, 5     brob, 6     cbrob, 8     Compare-methods, 9     Extract. brob, 10     getP, 11     glub, 12     length-methods, 14     Logic, 15     Math, 15     plot, 16     Print, 16     sum, 17  **Topic methods  Arith-methods, 4     Compare-methods, 9     length-methods, 14     trib-methods, 14     topic methods     Arith-methods, 4     Compare-methods, 9     length-methods, 14 **Topic methods     Arith-methods, 14 **Topic package     Brobdingnag-package, 2     cPair, ANY, ANY-method (brob-class), 7     cPair, ANY, glub-method (brob-class), 7     cPair, brob, brob-method (brob-class), 7     cPair, brob, complex-method	brob-class, 7	[,glub-method(Extract.brob), 10
*Topic math Arith-methods, 4 as.numeric, 5 brob, 6 cbrob, 8 Compare-methods, 9 Extract.brob, 10 getP, 11 glub, 12 length-methods, 14 Logic, 15 Math, 15 plot, 16 Print, 16 sum, 17 *Topic methods Arith-methods, 4 Compare-methods, 9 length-methods, 14 *Topic package Brobdingnag-package, 2 .cPair, ANY, brob-method (brob-class), 7 .cPair, ANY, glub-method (brob-class), 7 .cPair, brob, brob-method (brob-class), 7 .cPair, brob, complex-method (brob-class), 7 .cPair, brob, glub-method (glub-class), 13 .cPair, brob, glub-method (glub-class), 13 .cPair, complex, brob-method  (Arith-methods), 4 Arith, glub, brob-method (Arith-methods), 4 Arith, glub, brob-method (Arith-methods), 4 Arith, glub, brob-method (Arith-methods), 4 Arith, glub, complex-method (Arith-methods), 4 Arith, glub, brob-method (Arith-met	glub-class, 13	[.brob (Extract.brob), 10
Arith-methods, 4 as.numeric, 5 brob, 6 cbrob, 8 Compare-methods, 9 Complex, 9 Extract.brob, 10 getP, 11 glub, 12 length-methods, 14 Logic, 15 Math, 15 plot, 16 Print, 16 sum, 17 *Topic methods Arith-methods, 4 Compare-methods, 9 length-methods, 14 *Topic package Brobdingnag-package, 2 .cPair,ANY, hrob-method (brob-class), 7 .cPair,ANY, glub-method (brob-class), 7 .cPair, ANY, glub-method (brob-class), 7 .cPair, brob, complex-method	swift-class, 18	[.glub(Extract.brob), 10
as.numeric, 5 brob, 6 cbrob, 8 Compare-methods, 9 Complex, 9 Extract.brob, 10 getP, 11 glub, 12 length-methods, 14 Logic, 15 Math, 15 plot, 16 Print, 16 sum, 17 *Topic methods Arith-methods, 4 Compare-methods, 9 length-methods, 14 *Topic package Brobdingnag-package, 2 cPair,ANY, ANY-method (brob-class), 7 cPair, ANY, brob-method (brob-class), 7 cPair, brob, brob-method (brob-class), 7 cPair, brob, complex-method (brob-class), 7 cPair, brob, glub-method (glub-class), 13	*Topic <b>math</b>	<pre>[&lt;-,brob-method(Extract.brob), 10</pre>
brob, 6 cbrob, 8 Compare-methods, 9 Complex, 9 Extract.brob, 10 getP, 11 glub, 12 length-methods, 14 Logic, 15 Math, 15 plot, 16 Print, 16 sum, 17 *Topic methods Arith-methods, 4 Compare-methods, 9 length-methods, 14 *Topic package Brobdingnag-package, 2 .cPair, ANY, ANY-method (brob-class), 7 .cPair, ANY, glub-method (brob-class), 7 .cPair, brob, ANY-method (brob-class), 7 .cPair, brob, brob-method (brob-class), 7 .cPair, brob, brob-method (brob-class), 7 .cPair, brob, complex-method	Arith-methods, 4	<pre>[&lt;-,glub-method(Extract.brob), 10</pre>
cbrob, 8 Compare-methods, 9 Complex, 9 Extract.brob, 10 getP, 11 glub, 12 length-methods, 14 Logic, 15 Math, 15 plot, 16 Print, 16 sum, 17 *Topic methods Arith-methods, 4 Compare-methods, 9 length-methods, 14 *Topic package Brobdingnag-package, 2 .cPair, ANY, ANY-method (brob-class), 7 .cPair, ANY, glub-method (brob-class), 7 .cPair, brob, brob-method (brob-class), 7 .cPair, brob, complex-method	as.numeric,5	[ <brob(extract.brob), 10<="" td=""></brob(extract.brob),>
Compare-methods, 9 Complex, 9 Extract.brob, 10 getP, 11 glub, 12 length-methods, 14 Logic, 15 Math, 15 plot, 16 Print, 16 sum, 17 *Topic methods Arith-methods, 4 Compare-methods, 9 length-methods, 14 *Topic package Brobdingnag-package, 2 .cPair, ANY, ANY-method (brob-class), 7 .cPair, ANY, glub-method (brob-class), 7 .cPair, brob, brob-method (brob-class), 7 .cPair, brob, complex-method	brob, 6	<pre>[<glub (extract.brob),="" 10<="" pre=""></glub></pre>
Complex, 9 Extract.brob, 10 getP, 11 glub, 12 length-methods, 14 Logic, 15 Math, 15 plot, 16 Print, 16 sum, 17 *Topic methods Arith-methods, 14 *Topic package Brobdingnag-package, 2 .cPair,ANY,ANY-method (brob-class), 7 .cPair,ANY,glub-method (brob-class), 7 .cPair,brob, complex-method (brob-class), 7 .cPair,brob, complex-method (brob-class), 7 .cPair,brob, complex-method (brob-class), 7 .cPair,brob, complex-method (brob-class), 7 .cPair,brob, glub-method (glub-class), 13 .cPair,brob, complex-method (brob-class), 7 .cPair,brob, complex-method (brob-class), 7 .cPair,brob, complex-method (brob-class), 7 .cPair,brob, glub-method (glub-class), 13 .cPair,brob, complex-method (Arith-methods), 4 Arith, complex, glub-method (Arith-methods), 4 Arith, glub, ANY-method (Arith-methods), 4 Arith, glub, ANY-method (Arith-methods), 4 Arith, glub, brob-method (Arith-methods), 4 Arith, glub, complex-method (Arith-methods), 4 Arith, glub, complex-method (Arith-methods), 4 Arith, glub, glub-method (Arith-methods), 4 Arith, prob, and prob arith graph arith, brob, and arith, brob, complex arith, brob, brob-method (Arith-methods), 4 Arith, prob, prob-method (Arith-metho	cbrob, 8	
Extract.brob, 10 getP, 11 glub, 12 length-methods, 14 Logic, 15 Math, 15 plot, 16 Print, 16 sum, 17 *Topic methods Arith-methods, 14 *Topic package Brobdingnag-package, 2 .cPair, ANY, brob-method (brob-class), 7 .cPair, brob, ANY-method (brob-class), 7 .cPair, brob, brob-method (brob-class), 7 .cPair, brob, brob-method (brob-class), 7 .cPair, brob, complex-method (brob-class), 7 .cPair, brob, glub-method (glub-class), 13	Compare-methods, 9	
getP, 11 glub, 12 length-methods, 14 Logic, 15 Math, 15 plot, 16 Print, 16 sum, 17 *Topic methods Arith-methods, 4 Compare-methods, 9 length-methods, 14 *Topic package Brobdingnag-package, 2 .cPair, ANY, brob-method (brob-class), 7 .cPair, ANY, glub-method (brob-class), 7 .cPair, brob, brob-method (brob-class), 7 .cPair, brob, brob-method (brob-class), 7 .cPair, brob, complex-method (brob-class), 7 .cPair, brob, glub-method (glub-class), 13	Complex, 9	
getP, 11 glub, 12 length-methods, 14 Logic, 15 Math, 15 plot, 16 Print, 16 sum, 17 *Topic methods Arith-methods, 4 Compare-methods, 9 length-methods, 14 *Topic package Brobdingnag-package, 2 .cPair, ANY, ANY-method (brob-class), 7 .cPair, ANY, glub-method (brob-class), 7 .cPair, brob, ANY-method (brob-class), 7 .cPair, brob, brob-method (brob-class), 7 .cPair, brob, complex-method (brob-class), 7 .cPair, brob, glub-method (glub-class), 13 .cPair, complex, brob-method  Arith, glub, complex-method (Arith-methods), 4 Arith, glub, prob-method (Arith-methods), 4 Arith, glub, prob-method (Arith-methods), 4 Arith, brob, brob-method (Arit		
glub, 12 length-methods, 14 Logic, 15 Math, 15 plot, 16 Print, 16 sum, 17 *Topic methods Arith-methods, 4 Compare-methods, 9 length-methods, 14 *Topic package Brobdingnag-package, 2 .cPair, ANY, Brob-method (brob-class), 7 .cPair, ANY, glub-method (brob-class), 7 .cPair, brob, ANY-method (brob-class), 7 .cPair, brob, brob-method (brob-class), 7 .cPair, brob, complex-method		
length-methods, 14 Logic, 15 Math, 15 plot, 16 Print, 16 sum, 17 *Topic methods Arith-methods, 4 Compare-methods, 9 length-methods, 14 *Topic package Brobdingnag-package, 2 .cPair, ANY, Brob-method (brob-class), 7 .cPair, ANY, glub-method (brob-class), 7 .cPair, brob, brob-method (brob-class), 7 .cPair, brob, brob-method (brob-class), 7 .cPair, brob, complex-method (brob-class), 7 .cPair, brob, glub-method (brob-class), 7 .cPair, brob, brob-method (brob-class), 7 .cPair, brob, brob-method (brob-class), 7 .cPair, brob, complex-method (brob-class), 7 .cPair, brob, glub-method (glub-class), 13 .cPair, brob, glub-method (glub-class), 14 .cPair, brob, glub-method (glub-class), 14 .cPair, brob, glub-	_	
Logic, 15 Math, 15 plot, 16 Print, 16 sum, 17 *Topic methods Arith-methods, 4 Compare-methods, 14 *Topic package Brobdingnag-package, 2 .cPair, ANY, brob-method (brob-class), 7 .cPair, ANY, brob-method (brob-class), 7 .cPair, brob, complex-method (brob-class), 7 .cPair, brob, complex-method (brob-class), 7 .cPair, brob, complex-method (brob-class), 7 .cPair, brob, glub-method (glub-class), 13 .cPair, brob, glub-method (glub-class), 14 .cPair, brob, glub-method (glub-class), 14 .cPair, brob, glub-	_	
Math, 15 plot, 16 Print, 16 sum, 17 *Topic methods Arith-methods, 4 Compare-methods, 9 length-methods, 14 *Topic package Brobdingnag-package, 2 .cPair, ANY, Brub-method (Arith-methods), 4 Arith, brob, complex-method (Arith-methods), 4 Arith, brob, complex-method (Arith-methods), 4 Arith, brob, glub-method (Arith-methods), 4 Arith, brob, glub-method (Arith-methods), 4 Arith, brob, complex-method (Arith-methods), 4 Arith, brob, glub-method (Arith-methods), 4 Arith, complex, brob-method (Arith-methods), 4 Arith, brob, glub-method (Arith-methods), 4 Arith, brob, somplex-method (Arith-methods), 4 Arith, brob, somp	_	
plot, 16 Print, 16 sum, 17  *Topic methods Arith-methods, 4 Compare-methods, 9 length-methods, 14  *Topic package Brobdingnag-package, 2 .cPair, ANY, ANY-method (brob-class), 7 .cPair, brob, ANY-method (brob-class), 7 .cPair, brob, brob-method (brob-class), 7 .cPair, brob, complex-method (brob-class), 7 .cPair, brob, glub-method (glub-class), 13 .cPair, complex, brob-method  Arith, brob, complex-method (Arith-methods), 4  Arith, prob, and complex-method (Arith-methods), 4  Arith, brob, complex-method (Arith-methods), 4  Arith, complex, brob-method (Arit		
Print, 16 sum, 17  *Topic methods Arith-methods, 4 Compare-methods, 9 length-methods, 14  *Topic package Brobdingnag-package, 2 .cPair, ANY, ANY-method (brob-class), 7 .cPair, brob, brob-method (brob-class), 7 .cPair, brob, brob-method (brob-class), 7 .cPair, brob, brob-method (brob-class), 7 .cPair, brob, complex-method (brob-class), 7 .cPair, brob, complex-method (brob-class), 7 .cPair, brob, glub-method (glub-class), 13 .cPair, complex, brob-method  Arith, brob, complex-method (Arith-methods), 4 Arith, brob, glub-method (Arith-methods), 4 Arith, complex, glub-method (Arith-methods), 4 Arith, brob, complex-method (Arith-methods), 4 Arith, brob, complex-method (Arith-methods), 4 Arith, brob, glub-method (Arith-methods),		
*Topic methods Arith-methods, 4 Compare-methods, 9 length-methods, 14 *Topic package Brobdingnag-package, 2 .cPair, ANY, ANY-method (brob-class), 7 .cPair, brob, ANY-method (brob-class), 7 .cPair, brob, brob-method (brob-class), 7 .cPair, brob, complex-method (brob-class), 7 .cPair, brob, complex-method (brob-class), 7 .cPair, brob, complex-method (brob-class), 7 .cPair, brob, glub-method (glub-class), 7 .cPair, brob, glub-method (glub-class), 7 .cPair, brob, complex-method (brob-class), 7 .cPair, brob, glub-method (glub-class), 13 .cPair, complex, brob-method  4 Arith, brob, complex-method (Arith-methods), 4 Arith, complex, glub-method (Arith-methods), 4 Arith, glub, ANY-method (Arith-methods), 4 Arith, glub, complex-method (Arith-methods), 4 Arith, glub, complex-method (Arith-methods), 4 Arith, brob, complex-method (Arith-methods), 4 Arith, brob, glub-method (Arith-methods), 4 Arith, prob, missing-method (Arith-methods), 4 Arith, brob, missing-method (Arith-methods), 4 Arith, brob, missing-method (Arith-methods), 4 Arith, brob, glub-method (Arith-methods), 4 Arit		
*Topic methods Arith-methods, 4 Compare-methods, 9 length-methods, 14  *Topic package Brobdingnag-package, 2 .cPair,ANY,ANY-method (brob-class), 7 .cPair,brob,ANY-method (brob-class), 7 .cPair,brob,brob-method (brob-class), 7 .cPair,brob,complex-method (brob-class), 7 .cPair,brob,complex-method (brob-class), 7 .cPair,brob,glub-method (glub-class), 13 .cPair,brob,glub-method (glub-class), 13 .cPair,brob,glub-method (glub-class), 13 .cPair,brob,glub-method (glub-class), 13 .cPair,complex,brob-method  Arith,brob,glub-method (Arith-methods), 4  Arith,		•
Arith-methods, 4 Compare-methods, 9 length-methods, 14 *Topic package Brobdingnag-package, 2 .cPair, ANY, ANY-method (brob-class), 7 .cPair, brob, brob-method (brob-class), 7 .cPair, brob, brob-method (brob-class), 7 .cPair, brob, complex-method (brob-class), 7 .cPair, brob, glub-method (glub-class), 7 .cPair, brob, complex-method (brob-class), 7 .cPair, brob, glub-method (glub-class), 13 .cPair, brob, glub-method (Arith-methods), 4 Arith, brob, glub-method (Arith-methods),		
Compare-methods, 9 length-methods, 14 *Topic package Brobdingnag-package, 2 .cPair, ANY, ANY-method (brob-class), 7 .cPair, brob, brob-method (brob-class), 7 .cPair, brob, brob-method (brob-class), 7 .cPair, brob, complex-method (brob-class), 7 .cPair, brob, glub-method (glub-class), 13 .cPair, brob, glub-method (Arith-methods), 4 Arith, brob, glub-method (Arith-methods), 4 Arith, brob, missing-method (Arith-methods), 4 Arith, prob, missing-method (Arith-methods), 4 Arith, brob, glub-method (Arith-methods), 4 Arith, brob, glub-method (Arith-methods), 4 Arith, brob, glub-method (Arith-methods), 4 Arith, brob, missing-method (Arith-methods), 4 Arith, brob, missing-method (Arith-methods), 4 Arith, prob, missing-method (Arith-methods), 4 Arith, prob, missing-method (Arith-methods), 4 Arith, brob, glub-method (Arith-methods), 4 Arith, prob, missing-method (Arith-methods), 4 Arith, prob, missing-metho	÷	
length-methods, 14 *Topic package Brobdingnag-package, 2 .cPair, ANY, ANY-method (brob-class), 7 .cPair, ANY, glub-method (brob-class), 7 .cPair, brob, brob-method (brob-class), 7 .cPair, brob, complex-method (brob-class), 7 .cPair, brob, glub-method (glub-class), 13 .cPair, brob, glub-method (glub-class), 13 .cPair, brob, complex-method (brob-class), 7 .cPair, brob, glub-method (glub-class), 13 .cPair, brob, glub-method (Arith-methods), 4 Arith, glub, complex-method (Arith-methods), 4 Arith, glub, complex-method (Arith-methods), 4 Arith, glub, complex-method (Arith-methods), 4 Arith, glub, glub-method (Arith-methods), 4 Arith, glub, complex-method		
*Topic package  Brobdingnag-package, 2  .cPair, ANY, ANY-method (brob-class), 7  .cPair, ANY, glub-method (brob-class), 7  .cPair, brob, ANY-method (brob-class), 7  .cPair, brob, brob-method (brob-class), 7  .cPair, brob, complex-method  (brob-class), 7  .cPair, brob, glub-method (glub-class), 13  .cPair, brob, glub-method (Arith-methods), 4  Arith, complex, brob-method  (Arith-methods), 4  Arith, glub, ANY-method (Arith-methods), 4  Arith, glub, complex-method  (Arith-methods), 4  Arith, complex, brob-method  (Arith-methods), 4  Arith, glub, complex-method  (Arith-methods), 4  Arith, glub, glub-method (Arith-methods), 4  Arith, glub, glub-method (Arith-methods), 4  Arith, complex, brob-method	•	•
Brobdingnag-package, 2 .cPair, ANY, ANY-method (brob-class), 7 .cPair, ANY, glub-method (brob-class), 7 .cPair, brob, brob-method (brob-class), 7 .cPair, brob, complex-method	· · · · · · · · · · · · · · · · · · ·	
.cPair, ANY, ANY-method (brob-class), 7 .cPair, ANY, brob-method (brob-class), 7 .cPair, ANY, glub-method (glub-class), 13 .cPair, brob, ANY-method (brob-class), 7 .cPair, brob, brob-method (brob-class), 7 .cPair, brob, complex-method		
.cPair, ANY, brob-method (brob-class), 7 .cPair, ANY, glub-method (glub-class), 13 .cPair, brob, ANY-method (brob-class), 7 .cPair, brob, brob-method (brob-class), 7 .cPair, brob, complex-method		
.cPair, ANY, glub-method (glub-class), 13 .cPair, brob, ANY-method (brob-class), 7 .cPair, brob, brob-method (brob-class), 7 .cPair, brob, complex-method		
.cPair,brob,ANY-method(brob-class),7 .cPair,brob,brob-method(brob-class),7 .cPair,brob,complex-method		
.cPair,brob,brob-method (brob-class), 7 .cPair,brob,complex-method		
.cPair,brob,complex-method 4 (brob-class),7 Arith,glub,complex-method .cPair,brob,glub-method(glub-class),13 (Arith-methods),4 Arith,glub,glub-method(Arith-methods),		
(brob-class), 7		Arith, glub, brob-method (Arith-methods),
.cPair,brob,glub-method(glub-class), 13 (Arith-methods), 4 .cPair,complex,brob-method Arith,glub,glub-method(Arith-methods),		4
.cPair,complex,brob-method Arith,glub,glub-method(Arith-methods),		
,0 ,0 ,		
	(brob-class), 7	Arith, glub, glub-method (Arith-methods),
	.cPair,glub,ANY-method (glub-class), 13	Arith glub missing-method
	.cPair,glub,brob-method(glub-class), 13	
	.cPair,glub,glub-method (glub-class), 13	

20 INDEX

as.brob(brob), 6	Complex-methods (Complex), 9
as.complex(as.numeric),5	Conj(Complex), 9
as.complex,brob-method(as.numeric),5	Conj,brob-method(Complex),9
as.complex,glub-method(as.numeric),5	Conj,glub-method(Complex),9
as.glub(glub), 12	cos (Math), 15
as.numeric,5	cosh (Math), 15
as.numeric,brob-method(as.numeric),5	cumsum (Math), 15
as.numeric,glub-method(as.numeric),5	
asin (Math), 15	exp (Math), 15
asinh (Math), 15	Extract.brob, 10
atan (Math), 15	floor (Math), 15
atanh (Math), 15	11001 (riatil), 13
brob, 6, 11, 12	gamma (Math), 15
brob-class, 7	getP, 11
Brobdingnag (Brobdingnag-package), 2	<pre>getP, brob-method (brob-class), 7</pre>
Brobdingnag-package, 2	getX (getP), 11
bi obdingnag package, 2	<pre>getX,brob-method(brob-class),7</pre>
cBrob (cbrob), 8	glub, 7, 12
cbrob, 8	glub-class, 13
ceiling (Math), 15	T (0 1 ) 0
coerce, brob, complex-method	Im (Complex), 9
(as.numeric), 5	Im, brob-method (Complex), 9
coerce, brob, numeric-method	Im, glub-method (Complex), 9
(as.numeric), 5	Im<- (Complex), 9
coerce,glub,complex-method	Im<-, brob-method (Complex), 9
(as.numeric), 5	<pre>Im&lt;-,glub-method (Complex), 9</pre>
coerce,glub,numeric-method	is.brob (brob), 6
(as.numeric), 5	is.glub(glub), 12 is.na, <i>17</i>
Compare, ANY, brob-method	15.1ld, 17
(Compare-methods), 9	length (length-methods), 14
Compare,ANY,glub-method	length, brob-method (length-methods), 14
(Compare-methods), 9	length,glub-method (length-methods), 14
Compare,brob,ANY-method	length-methods, 14
(Compare-methods), 9	lgamma (Math), 15
Compare,brob,brob-method	log (Math), 15
(Compare-methods), 9	Logic, 15
Compare,brob,glub-method	Logic, ANY, swift-method (Logic), 15
(Compare-methods), 9	Logic, swift, ANY-method (Logic), 15
Compare,glub,ANY-method	Logic, swift, swift-method (Logic), 15
(Compare-methods), 9	logic.brob(Logic), 15
Compare,glub,brob-method	
(Compare-methods), $9$	Math, 15
Compare,glub,glub-method	Math, brob-method (Math), 15
(Compare-methods), 9	Math,glub-method(Math),15
Compare-methods, 9	max (sum), 17
Complex, 9	min (sum), 17
Complex, brob-method (Complex), 9	Mod (Complex), 9
Complex,glub-method(Complex),9	Mod, brob-method (Complex), 9

INDEX 21

```
Mod, glub-method (Complex), 9
plot, 16
plot, ANY, brob-method (plot), 16
plot, ANY, glub-method (plot), 16
plot, brob, ANY-method (plot), 16
plot, brob, missing-method (plot), 16
plot, brob-method (plot), 16
plot, glub, ANY-method (plot), 16
plot, glub, missing-method (plot), 16
plot, glub-method (plot), 16
Print, 16
print.brob (Print), 16
print.glub (Print), 16
prod (sum), 17
range (sum), 17
Re (Complex), 9
Re, brob-method (Complex), 9
Re, glub-method (Complex), 9
Re<- (Complex), 9
Re<-,glub-method(Complex),9
show, brob-method (Print), 16
show, glub-method (Print), 16
sign<- (getP), 11
sign<-,brob-method(brob-class), 7</pre>
sin (Math), 15
sinh (Math), 15
sqrt (Math), 15
sqrt, brob-method (Math), 15
sqrt,glub-method (Math), 15
sum, 17
Summary, brob-method (sum), 17
Summary, glub-method (sum), 17
swift-class, 18
tan (Math), 15
tanh (Math), 15
trunc (Math), 15
```