Package 'schoolmath'

July 8, 2026

Type Package

Title Functions and Datasets for Math Used in School

Description Contains functions and datasets for math taught in school. A main focus is set to prime-calculation.

Version 0.5.0

Date 2026-07-07

License GPL (>= 2)

Depends R (>= 4.0)

Encoding UTF-8

Language en-US

LazyData true

LazyDataCompression xz

NeedsCompilation yes

RoxygenNote 7.3.3

LinkingTo Rcpp

Imports Rcpp

Author Jörg große Schlarmann [aut, cre]

Maintainer Jörg große Schlarmann <schlarmann@produnis.de>

Repository CRAN

Date/Publication 2026-07-08 00:40:02 UTC

Contents

`cancel.fraction` *cancel a fraction to the smallest numbers*

Description

cancel a fraction to the smallest numbers

Usage

```
cancel.fraction(numerator, denominator)
```

Arguments

`numerator`	the fraction's numerator
`denominator`	the fraction's denominator #'

Value

An integer vector of length 2 containing the reduced numerator and denominator.

Examples

```
cancel.fraction(40,15)
cancel.fraction(42, 56)
```

`decimal2fraction` *convert a decimal-number into fraction*

Description

convert a decimal-number into fraction

Usage

```
decimal2fraction(decimal, period = 0)
```

Arguments

`decimal`	the decimal number to be converted, given without an repeating ending
`period`	if the decimal places have an repeating ending (period), set the period here. See examples. #'

Value

An integer vector of length 2 containing the numerator and denominator

Examples

```
## converting 23.4323
decimal2fraction(23.4323)

## converting a number with decimal period, e.g. 12.12344444444444444444
decimal2fraction(12.123, 4)
```

---

gcd *Greatest common divisor of two numbers*

Description

Greatest common divisor of two numbers

Usage

```
gcd(x, y)
```

Arguments

x	first number
y	second number #'

Value

numeric greatest common divisor

Examples

```
gcd(42, 56)
```

is.prim *Check whether a vector contains prime numbers*

Description

Check whether a vector contains prime numbers

Usage

```
is.prim(x)
```

Arguments

x	The number or vector to check.

Value

A logical vector indicating whether values are prime.

Examples

```
is.prim(8)
is.prim(11)

x <- c(1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11)
is.prim(x)
```

prime.factor *This function calculates the prime-factors of a number*

Description

This function calculates the prime-factors of a number

Usage

```
prime.factor(n)
```

Arguments

n	the number to be checked #'

Value

a vector with the prime factors

Examples

```
prime.factor(21)
prime.factor(100)
```

`primes` *generate prime-numbers in a range from START to END*

Description

generate prime-numbers in a range from START to END

Usage

```
primes(start = 1, end = 9999, live = FALSE)
```

Arguments

`start`	the number to start from
`end`	the number to end
`live`	should primes additionally be printed out to stdout when found? (default is FALSE)

Value

a vector of prime numbers

Examples

```
primes(12,150) # list prime-numbers between 12 and 150
```

`primlist` *A vector containing primes between 1 and 99,999,999*

Description

Contains primes between 1 and 99,999,999

Usage

```
data(primlist)
```

Format

A vector containing primes between 1 and 99,999,999

Details

Variables in the dataset:

- primlist. A vector containing primes between 1 and 99,999,999

`scm` *calculating the smallest common multiple of two numbers*

Description

calculating the smallest common multiple of two numbers

Usage

```
scm(x, y)
```

Arguments

`x`	first number
`y`	second number #'

Value

numeric least common multiple

Examples

```
scm(3528, 3780)
```

Index