
codejamhelpers Documentation

Release 0.1.1

Jun 01, 2018

Contents

1	Indices and tables	3
	Python Module Index	5

A library of helper functions useful for solving Google Code Jam problems.

Contents:

class `codejamhelpers.Primes` (*frontier=None*)

The prime numbers, prepopulated up to *frontier*, with lazy evaluation beyond.

__contains__ (*n*)

 Test whether *n* is prime

__getitem__ (*i*)

 The *i*th prime

count (*x*)

 Count the number of primes less than or equal to *x*

`codejamhelpers.binary_search` (*f*, *t*)

Given an increasing function *f*, find the greatest non-negative integer *n* such that $f(n) \leq t$. If $f(n) > t$ for all $n \geq 0$, return `None`.

`codejamhelpers.powers_of_two` ()

Powers of 2, from 1

`codejamhelpers.minimise_convex` (*f*)

Given a U-shaped (convex and eventually increasing) function *f*, find its minimum over the non-negative integers. That is *m* such that $f(m) \leq f(n)$ for all *n*. If there exist multiple solutions, return the largest. Uses binary search on the derivative.

`codejamhelpers.minimise_convex2` (*f*)

Given a U-shaped (convex and eventually increasing) function *f*, find its minimum over the non-negative integers. That is *m* such that $f(m) \leq f(n)$ for all *n*. If there exist multiple solutions, return the largest. Uses ternary search.

`codejamhelpers.kth_root` (*n*, *k*)

Calculate the greatest non-negative integer *r* such that $r^k \leq n$.

`codejamhelpers.trials` (*P*)

Given the individual probabilities P_i of *n* independent trials, calculate the probabilities Q_k that exactly $0 \leq k \leq n$ are successful.

CHAPTER 1

Indices and tables

- `genindex`
- `modindex`
- `search`

C

`codejamhelpers`, [1](#)

Symbols

`__contains__()` (codejamhelpers.Primes method), 1
`__getitem__()` (codejamhelpers.Primes method), 1

B

`binary_search()` (in module codejamhelpers), 1

C

`codejamhelpers` (module), 1
`count()` (codejamhelpers.Primes method), 1

K

`kth_root()` (in module codejamhelpers), 1

M

`minimise_convex()` (in module codejamhelpers), 1
`minimise_convex2()` (in module codejamhelpers), 1

P

`powers_of_two()` (in module codejamhelpers), 1
`Primes` (class in codejamhelpers), 1

T

`trials()` (in module codejamhelpers), 1