Lemon

Lemon Package Documentation

- Methods

MEMBERS

L.A. Guy © 2021

R. Surio Jr. © 2021

M.A. Sustento © 2021

T.J. Vallarta © 2021

Lemon

METHODS

- B_force()
- B_forcex()
- Newton()
- Bisection()
- Falsi()
- Secant()

Lemon

lemon.b_force(func, n_roots, epochs, x0, x1, tol=1e-5)

Definition: Returns the roots of an equation given in *func* using the brute force method (simple iterative method).

Parameters: **func:** ***array_like***

Input arrays with equation.

n_roots: ***int***

The desired number of roots.

epochs: ***int***

The desired number of iterations.

x0: ***int***

Starting value.

x1: ***int***

Starting value.

tol: ***float***

Tolerance.

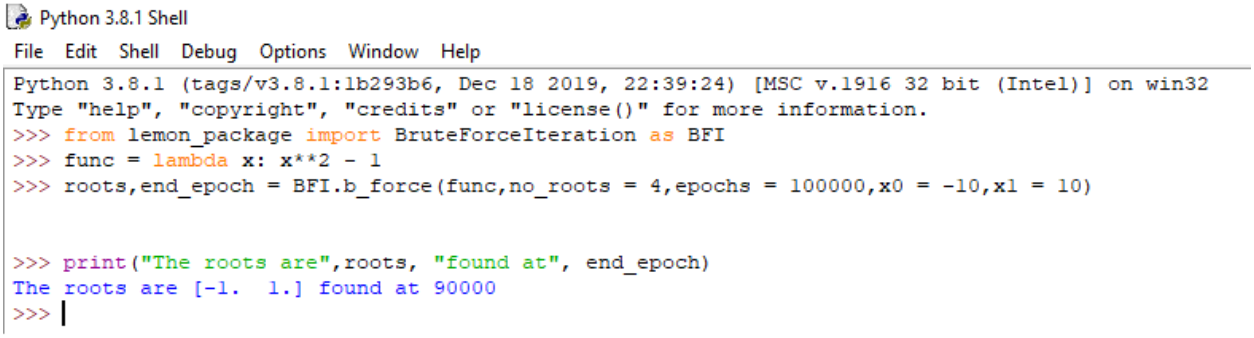
Returns: **Roots:** ***List***

A list containing the roots of the equation.

Epochs: ***int***

An integer containing the number of iterations where the root is found.

Examples:



Lemon

lemon.b_forcex(func, n_roots, epochs, x0 = 0)

Definition: Returns the roots of an equation given in *func* using the brute force method (in terms of x).

Parameters: **func:** ***array_like***

Input arrays with equation.

n_roots: ***int***

The desired number of roots.

epochs: ***int***

The desired number of iterations.

x0: ***int***

Starting value.

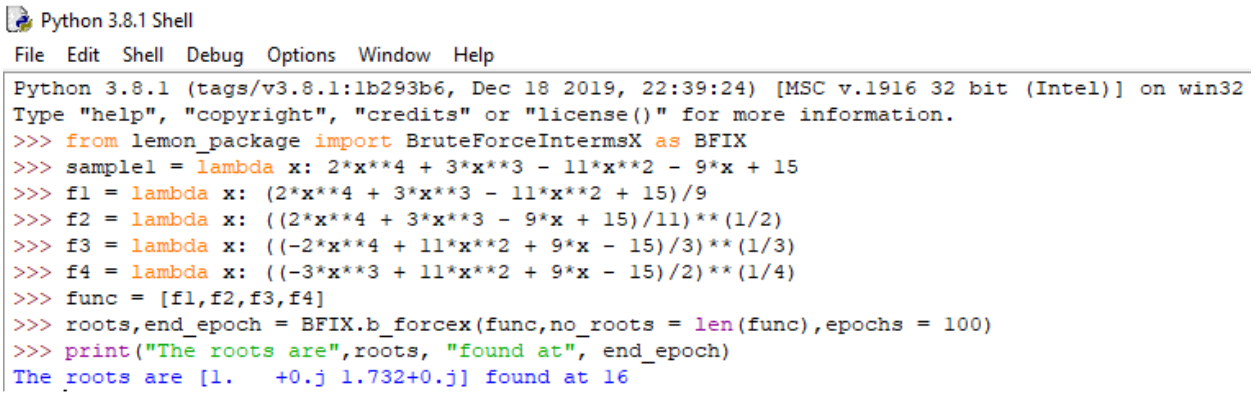
Returns: **Roots:** ***List***

A list containing the roots of the equation.

Epochs: ***int***

An integer containing the number of iterations where the root is found.

Examples:



Lemon

lemon.newton(func, n_roots, epochs, tol = 1.0e-05, inits)

Definition: Returns the roots of an equation given in *func* using the newton-raphson method.

Parameters:

func: ***array_like***

Input arrays with equation.

n_roots: ***int***

The desired number of roots.

epochs: ***int***

The desired number of iterations.

tol: ***float***

Tolerance**.**

inits: ***int***

Return values within a given interval.

Returns:

Roots: ***List***

A list containing the roots of the equation.

Epochs: ***int***

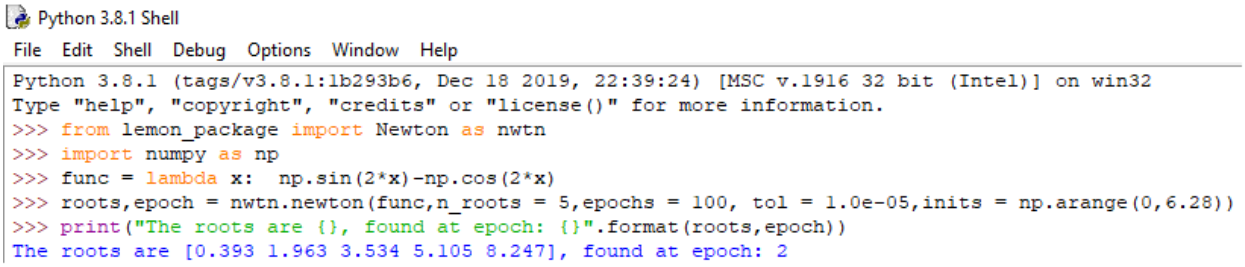
An integer containing the number of iterations where the root is found.

Raises:

RuntimeWarning

When encountered invalid value in log or double_scalars.

Examples:



Lemon

lemon.bisection(func, i1, i2, n_roots, epochs, n_move, tol = 1.0e-06)

Definition: Returns the roots of an equation given in *func* using the bisection method.

Parameters: **func:** ***array_like***

Input arrays with equation.

i1*: int*

Intervals where the root is expected.

i2*: int*

Intervals where the root is expected.

n_roots: ***int***

The desired number of roots.

epochs: ***int***

The desired number of iterations.

n_move: ***int***

Increments or decrements of moving interval.

tol: ***float***

Tolerance**.**

Returns: **Roots:** ***List***

A list containing the roots of the equation.

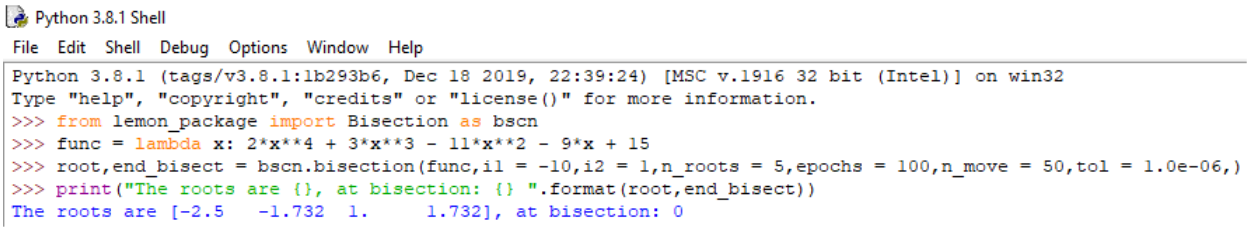
Epochs: ***int***

An integer containing the number of iterations where the root is found.

Raises: **RuntimeWarning**

When encountered invalid value in log and divide by zero in log.

Examples:



Lemon

lemon.falsi(func, a, b, n_roots, epochs, n_move, tol = 1.0e-06,)

Definition: Returns the roots of an equation given in *func* using the regula falsi (false position) method.

Parameters: **func: *array_like***

Input arrays with equation.

a*: int*

Intervals where the root is expected.

b*: int*

Intervals where the root is expected.

n_roots: *int*

The desired number of roots.

epochs: *int*

The desired number of iterations.

tol: *float*

Tolerance**.**

n_move: *int*

Increments or decrements of moving interval.

Returns: **Roots: *List***

A list containing the roots of the equation.

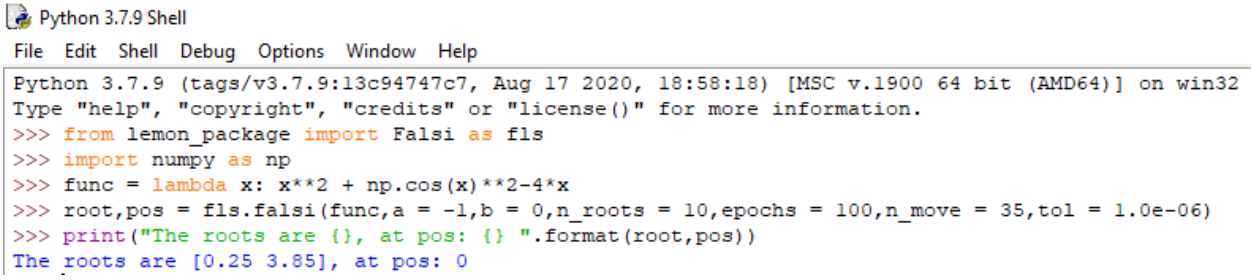
Epochs: *int*

An integer containing the number of iterations where the root is found.

Raises: **RuntimeWarning**

When encountered invalid value in log.

Examples:



Lemon

lemon.secant(func, a, b, n_roots, epochs, n_move, tol = 1.0e-06)

Definition: Returns the roots of an equation given in *func* using the secant method.

Parameters: **func: *array_like***

Input arrays with equation.

a*: int*

Intervals where the root is expected.

b*: int*

Intervals where the root is expected.

n_roots: *int*

The desired number of roots.

epochs: *int*

The desired number of iterations.

n_move: *int*

Increments or decrements of moving interval.

tol: *float*

Tolerance**.**

Returns: **Roots: *List***

A list containing the roots of the equation.

Epochs: *int*

An integer containing the number of iterations where the root is found.

Raises: **RuntimeWarning**

When encountered invalid value in log.

Examples: