Reference Matalg

NAME

Matalg - # Matalg.py - module for Matrix class.

CLASSES

builtins.list(builtins.object)

Matrix

class Matrix(builtins.list)

"Matrix" creates a (m x n) matrix of float 0.'s

Method resolution order:

Matrix builtins.list builtins.object

Methods defined here:

Private Methods

Private methods should not be accessed from the programs.

__add__(self, other)

Matrix addition.

__eq__(self, other)

Matrix equality

__getitem__(self, key)

get index value.

__init__(self, m=1, n=1)

Create zero matrix of m = no of rows, n = no of columns.

__invert__(self)

[~] Matrix inversion).

__mul__(self, other)

Matrix multiplication: [*] when both self and other are matrices. When one is a scalar and the other is a matrix, scalar multiplicatin of a matrix.

__pow__(self, other)

[**] Equation solver x = self * rhs).

__rmul__(self, other)

number * matrix --> matrix

__setitem__(self, key, value)

Set index value.

__str__(self)

Matrix string format for print function.

__sub__(self, other)

Matrix subtraction.

Public Methods

Public mathods are for general use.

matadd(self, other)

return (mat add) = self + other

matbinop(self, other)

Mat addition and subtraction.

matcopy(self)

Creates and returns a copy of the matrix.

matequal(self, other)

Matrices are equal, return True or False.

matinvert(self)

amat.matinvert() --> inverse of amat.

matmult(self, other)

self.matmult(other) --> matrix product self x other.

matprint(self)

Usage: to print amat issue amat.matprint() .

matsub(self, other)

return (mat Subtract) = self - other.

mattranspose(self)

self.mattranspose --> returns transpose (self unchanged).

matunit(self)

Make self the unit matrix (in place).

neatprint(self, prnt=<function printline>, LineLen=5)

prinline = the line printing function. Neatly prints matrix self of size (m x n). Line length variable, default is 5.

scalarmult(self, factor)

Multiply matrix by scalar (in place).

solve(self, other)

other is rhs and is returned as solution. Partial pivoting.

solve_obsolete(self, rhs)

This is an obsolete method that may be useful for debugging; then may be removed.

Methods inherited from builtins.list:

__contains__(...)

x.__contains__(y) <==> y in x

__delitem__(...)

x.__delitem__(y) <==> del x[y]

__ge__(...)

x.__ge__(y) <==> x>=y

__getattribute__(...)

x.__getattribute__('name') <==> x.name

__gt__(...)

x.__gt__(y) <==> x>y

__iadd__(...)

x.__iadd__(y) <==> x+=y

__imul__(...)

  x.__imul__(y) <==> x*=y

__iter__(...)

  x.__iter__() <==> iter(x)

__le__(...)

  x.__le__(y) <==> x<=y

__len__(...)

  x.__len__() <==> len(x)

__lt__(...)

  x.__lt__(y) <==> x<y

__ne__(...)

  x.__ne__(y) <==> x!=y

__repr__(...)

  x.__repr__() <==> repr(x)

__reversed__(...)

  L.__reversed__() -- return a reverse iterator over the list

__sizeof__(...)

  L.__sizeof__() -- size of L in memory, in bytes

append(...)

  L.append(object) -- append object to end

count(...)

  L.count(value) -> integer -- return number of occurrences of value

extend(...)

  L.extend(iterable) -- extend list by appending elements from the iterable

index(...)

  L.index(value, [start, [stop]]) -> integer -- return first index of value. Raises ValueError if the value is not present.

insert(...)

  L.insert(index, object) -- insert object before index

pop(...)

  L.pop([index]) -> item -- remove and return item at index (default last). Raises IndexError if list is empty or index is out of range.

remove(...)

  L.remove(value) -- remove first occurrence of value. Raises ValueError if the value is not present.

reverse(...)

  L.reverse() -- reverse *IN PLACE*

sort(...)

  L.sort(key=None, reverse=False) -- stable sort *IN PLACE*

Inherited Attributes

Data and other attributes inherited from builtins.list:

__new__ = <built-in method __new__ of type object>

  T.__new__(S, ...) -> a new object with type S, a subtype of T

FUNCTIONS

enterdata(m, n, datalist, autoprint=True)

   datalist --> create matrix and enter data into matrix.

find_max(lst)

   Find maximum *absolute* value of entries in a list, together with its offsetfrom origin.

mkunitmat(m)

   m --> make (mxm) unit matrix and print by default.

mkzeromat(m, n, autoprint=False)

   m, n --> make (mxn) zero matrix and print by default.

printline(line)

   Function to simmulate appending to a plainText widget.

printm(fo, printline, mat, LineLen=5)

   This is obsolete - kept as source material only.

printmat(message, mat)

   convenience method.

VERSION

0.0.9

FILE

/dat/work/py3/MatAlg/matalg/Matalg.py