Reference Matalg27

NAME

Matalg27

FILE

/usr/local/lib/python2.7/dist-packages/matalg27/Matalg27.py

DESCRIPTION

Matalg27.py - module for Matrix class. Version for Python2.7.

CLASSES

Builtins List

__builtin__.list(__builtin__.object)

Matrix

class Matrix(__builtin__.list)

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

Method resolution order:

Matrix __builtin__.list __builtin__.object

Private Methods

Methods defined here:

__add__(self, other)

__eq__(self, other)

__getitem__(self, key)

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

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 for x = self * rhs).

__rmul__(self, other)

number * matrix --> matrix

__setitem__(self, key, value)

__str__(self)

__sub__(self, other)

Public Methods

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 unit matrix and return it.

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). ("in place" is not a "good place"... better return the result in a new matrix!)

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 to be removed.

transp(self, other)

Data descriptors defined here:

__**dict**__
dictionary for instance variables (if defined)

__**weakref**__
list of weak references to the object (if defined)

Inherited Methods.

Methods inherited from __builtin__.list:

__**contains**__**(...)**
x.__contains__(y) <==> y in x

__delitem__(...)

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

__delslice__(...)

x.__delslice__(i, j) <==> del x[i:j]
Use of negative indices is not supported.

__ge__(...)

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

__getattribute__(...)

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

__getslice__(...)

x.__getslice__(i, j) <==> x[i:j]
Use of negative indices is not supported.

__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

__setslice__(...)

x.__setslice__(i, j, y) <==> x[i:j]=y
Use of negative indices is not supported.

__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(cmp=None, key=None, reverse=False) -- stable sort *IN PLACE*; cmp(x, y) -> -1, 0, 1

Data and other attributes inherited from __builtin__.list:

__hash__ = None

__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 store matrix and enter data into store.

find_max(lst)

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

mkunitmat(m, autoprint=True)

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.

printmat(message, mat)

Convenience method. Non-essential.

DATA

__version__ = '0.1.0'

VERSION

0.1.0