Contents

Chapter 1

Classes

1.1 vector – vector object and arithmetic

- **Classes**
  - **Vector**
- **Functions**
  - **innerProduct**

This module provides an exception class.

VectorSizeError : Report vector size is invalid. (Mainly for operations with two vectors.)

1.1.1 Vector – vector class

Vector is a class for vector.

Initialize (Constructor)

Vector(compo: ***list***) → ***Vector***

Create Vector object from compo. compo must be a list of elements which are an integer or an instance of **RingElement**.

Attribute

compo :
It expresses component of vector.

Operations

Note that index is 1-origin, which is standard in mathematics field.

operator	explanation
u+v	Vector sum.
u-v	Vector subtraction.
A*v	Multiplication vector with matrix
a*v	or scalar multiplication.
v//a	Scalar division.
v%n	Reduction each elements of **compo**
-v	element negation.
u==v	equality.
u!=v	inequality.
v[i]	Return the coefficient of i-th element of Vector.
v[i] = c	Replace the coefficient of i-th element of Vector by c.
len(v)	return length of **compo**.
repr(v)	return representation string.
str(v)	return string of **compo**.

Examples

```
>>> A = vector.Vector([1, 2])
>>> A
Vector([1, 2])
>>> A.compo
[1, 2]
```

```
>>> B = vector.Vector([2, 1])
>>> A + B
Vector([3, 3])
>>> A % 2
Vector([1, 0])
>>> A[1]
1
>>> len(B)
2
```

Methods

1.1.1.1 copy – copy itself

copy(`self`) → *Vector*

Return copy of `self`.

1.1.1.2 set – set other compo

set(`self`, `compo`: *list*) → (None)

Substitute **compo** with `compo`.

1.1.1.3 indexOfNoneZero – first non-zero coordinate

indexOfNoneZero(`self`) → integer

Return the first index of non-zero element of `self`.**compo**.

†Raise ValueError if all elements of **compo** are zero.

1.1.1.4 toMatrix – convert to Matrix object

toMatrix(`self`, `as_column`: *bool*=False) → *Matrix*

Return **Matrix** object using **createMatrix** function.

If `as_column` is True, create the column matrix with `self`. Otherwise, create the row matrix.

Examples

```
>>> A = vector.Vector([0, 4, 5])
>>> A.indexOfNoneZero()
2
>>> print A.toMatrix()
0 4 5
>>> print A.toMatrix()
```

```
0
4
5
```

1.1.2 innerProduct(function) – inner product

innerProduct(bra: ***Vector***, ket: ***Vector***) → ***RingElement***

Return the inner product of bra and ket.

The function supports Hermitian inner product for elements in the complex number field.

†Note that the returned value depends on type of elements.

Examples

```
>>> A = vector.Vector([1, 2, 3])
>>> B = vector.Vector([2, 1, 0])
>>> vector.innerProduct(A, B)
4
>>> C = vector.Vector([1+1j, 2+2j, 3+3j])
>>> vector.innerProduct(C, C)
(28+0j)
```