Contents

Chapter 1

Classes

1.1 intresidue – integer residue

intresidue module provides integer residue classes or $\mathbf{Z}/m\mathbf{Z}$.

- **Classes**
  - **IntegerResidueClass**
  - **IntegerResidueClassRing**

1.1.1 IntegerResidueClass – integer residue class

This class is a subclass of **CommutativeRingElement**.

Initialize (Constructor)

IntegerResidueClass(representative: ***integer***, **modulus:** ***integer***)
→ ***Integer***

Create a residue class of modulus with residue representative.
modulus must be positive integer.

Operations

operator	explanation
`a+b`	addition.
`a-b`	subtraction.
`a*b`	multiplication.
`a/b`	division.
`a**i,pow(a,i)`	power.
`-a`	negation.
`+a`	make a copy.
`a==b`	equality or not.
`a!=b`	inequality or not.
`repr(a)`	return representation string.
`str(a)`	return string.

Methods

1.1.1.1 getRing – get ring object

getRing(self) → *IntegerResidueClassRing*

Return a ring to which it belongs.

1.1.1.2 getResidue – get residue

getResidue(self) → *integer*

Return the value of residue.

1.1.1.3 getModulus – get modulus

getModulus(self) → *integer*

Return the value of modulus.

1.1.1.4 inverse – inverse element

inverse(self) → *IntegerResidueClass*

Return the inverse element if it is invertible. Otherwise raise ValueError.

1.1.1.5 minimumAbsolute – minimum absolute representative

minimumAbsolute(self) → Integer

Return the minimum absolute representative integer of the residue class.

1.1.1.6 minimumNonNegative – smallest non-negative representative

minimumNonNegative(self) → Integer

Return the smallest non-negative representative element of the residue class.
†this method has an alias, named toInteger.

1.1.2 IntegerResidueClassRing – ring of integer residue

The class is for rings of integer residue classes.

This class is a subclass of **CommutativeRing**.

Initialize (Constructor)

IntegerResidueClassRing(`modulus`: *integer*) → *IntegerResidueClassRing*

Create an instance of IntegerResidueClassRing. The argument `modulus` = m specifies an ideal $m\mathbb{Z}$.

Attributes

zero :
It expresses The additive unit 0. (read only)

one :
It expresses The multiplicative unit 1. (read only)

Operations

operator	explanation
`R==A`	ring equality.
`card(R)`	return cardinality. See also **compatibility** module.
`e in R`	return whether an element is in or not.
`repr(R)`	return representation string.
`str(R)`	return string.

Methods

1.1.2.1 createElement – create IntegerResidueClass object

createElement(self, seed: ***integer*****)** → ***Integer***

Return an IntegerResidueClass instance with seed.

1.1.2.2 isfield – field test

isfield(self) → ***bool***

Return True if the modulus is prime, False if not. Since a finite domain is a field, other ring property tests are merely aliases of isfield; they are isdomain, iseuclidean, isnoetherian, ispid, isufd.

1.1.2.3 getInstance – get instance of IntegerResidueClassRing

getInstance(cls, modulus: ***integer*****)** → ***IntegerResidueClass***

Return an instance of the class of specified modulus. Since this is a class method, use it as:

```
IntegerResidueClassRing.getInstance(3)
```

to create a $\mathbb{Z}/3\mathbb{Z}$ object, for example.

Bibliography