

Sage Quick Reference: Abstract Algebra

B. Balof, T. W. Judson, D. Perkinson, R. Potluri

version 1.0, Sage Version 5.0.1

latest version: <http://wiki.sagemath.org/quickref>

GNU Free Document License, extend for your own use

Based on work by P. Jipsen, W. Stein, R. Beezer

Basic Help

`com<tab>` complete *command*

`a.<tab>` all methods for object *a*

`<command>?` for summary and examples

`<command>??` for complete source code

`*foo*` list all commands containing *foo*

`_` underscore gives the previous output

www.sagemath.org/doc/reference online reference

www.sagemath.org/doc/tutorial online tutorial

`load foo.sage` load commands from the file *foo.sage*

`attach foo.sage`

loads changes to *foo.sage* automatically

Lists

`L = [2,17,3,17]` an ordered list

`L[i]` the *i*th element of *L*

Note: lists begin with the 0th element

`L.append(x)` adds *x* to *L*

`L.remove(x)` removes *x* from *L*

`L[i:j]` the *i*-th through (*j* - 1)-th element of *L*

`range(a)` list of integers from 0 to *a* - 1

`range(a,b)` list of integers from *a* to *b* - 1

`[a..b]` list of integers from *a* to *b*

`range(a,b,c)`

every *c*-th integer starting at *a* and less than *b*

`len(L)` length of *L*

`M = [i^2 for i in range(13)]`

list of squares of integers 0 through 12

`N = [i^2 for i in range(13) if is_prime(i)]`

list of squares of prime integers between 0 and 12

`M + N` the concatenation of lists *M* and *N*

`sorted(L)` a sorted version of *L* (*L* is not changed)

`L.sort()` sorts *L* (*L* is changed)

`set(L)` an unordered list of unique elements

Programming Examples

Print the squares of the integers 0, ..., 14:

```
for i in range(15):
    print i^2
```

Print the squares of those integers in {0, ..., 14} that are relatively prime to 15:

```
for i in range(13):
    if gcd(i,15)==1:
        print i^2
```

Preliminary Operations

`a = 3; b = 14`

`gcd(a,b)` greatest common divisor *a, b*

`xgcd(a,b)`

triple (*d, s, t*) where $d = sa + tb$ and $d = \gcd(a, b)$

`next_prime(a)` next prime after *a*

`previous_prime(a)` prime before *a*

`prime_range(a,b)` primes *p* such that $a \leq p < b$

`is_prime(a)` is *a* a prime?

`b % a` the remainder of *b* upon division by *a*

`a.divides(b)` does *a* divide *b*?

Group Constructions

Permutation multiplication is left-to-right.

`G = PermutationGroup([(1,2,3),(4,5)],[(3,4)])`

perm. group with generators (1, 2, 3)(4, 5) and (3, 4)

`G = PermutationGroup(["(1,2,3)(4,5)","(3,4)"])`

alternative syntax for defining a permutation group

`S = SymmetricGroup(4)` the symmetric group, S_4

`A = AlternatingGroup(4)` alternating group, A_4

`D = DihedralGroup(5)` dihedral group of order 10

`Ab = AbelianGroup([0,2,6])` the group $\mathbb{Z} \times \mathbb{Z}_2 \times \mathbb{Z}_6$

`Ab.0, Ab.1, Ab.2` the generators of *Ab*

`a,b,c = Ab.gens()`

shorthand for `a = Ab.0; b = Ab.1; c = Ab.2`

`C = CyclicPermutationGroup(5)`

`Integers(8)` the group $\mathbb{Z}_8$

`GL(3,QQ)` general linear group of 3×3 matrices

`m = matrix(QQ,[[1,2],[3,4]])`

`n = matrix(QQ,[[0,1],[1,0]])`

`MatrixGroup([m,n])`

the (infinite) matrix group with generators *m* and *n*

`u = S([(1,2),(3,4)]); v = S((2,3,4))` elements of *S*

`S.subgroup([u,v])`

the subgroup of *S* generated by *u* and *v*

`S.quotient(A)` the quotient group *S/A*

`A.cartesian_product(D)` the group $A \times D$

`A.intersection(D)` the intersection of groups *A* and *D*

`D.conjugate(v)` the group $v^{-1}Dv$

`S.sylow_subgroup(2)` a Sylow 2-subgroup of *S*

`D.center()` the center of *D*

`S.centralizer(u)` the centralizer of *x* in *S*

`S.centralizer(D)` the centralizer of *D* in *S*

`S.normalizer(u)` the normalizer of *x* in *S*

`S.normalizer(D)` the normalizer of *D* in *S*

`S.stabilizer(3)` subgroup of *S* fixing 3

Group Operations

`S = SymmetricGroup(4); A = AlternatingGroup(4)`

`S.order()` the number of elements of *S*

`S.gens()` generators of *S*

`S.list()` the elements of *S*

`S.random_element()` a random element of *S*

`u*v` the product of elements *u* and *v* of *S*

`v^(-1)*u^3*v` the element $v^{-1}u^3v$ of *S*

`u.order()` the order of *u*

`S.subgroups()` the subgroups of *S*

`S.normal_subgroups()` the normal subgroups of *S*

`A.cayley_table()` the multiplication table for *A*

`u in S` is *u* an element of *S*?

`u.word_problem(S.gens())`

write *u* as a product of the generators of *S*

`A.is_abelian()` is *A* abelian?

`A.is_cyclic()` is *A* cyclic?

`A.is_simple()` is *A* simple?

`A.is_transitive()` is *A* transitive?

`A.is_subgroup(S)` is *A* a subgroup of *S*?

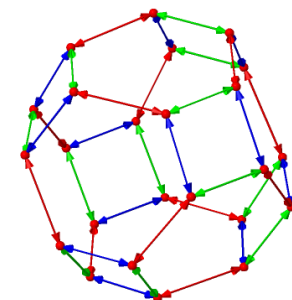
`A.is_normal(S)` is *A* a normal subgroup of *S*?

`S.cosets(A)` the right cosets of *A* in *S*

`S.cosets(A,'left')` the left cosets of *A* in *S*

`g = S.cayley_graph()` Cayley graph of *S*

`g.show3d(color_by_label=True, edge_size=0.01, vertex_size=0.03)` see below:



Ring and Field Constructions

ZZ integral domain of integers, $\mathbb{Z}$
Integers(7) ring of integers mod 7, $\mathbb{Z}_7$
QQ field of rational numbers, $\mathbb{Q}$
RR field of real numbers, $\mathbb{R}$
CC field of complex numbers, $\mathbb{C}$
RDF real double field, inexact
CDF complex double field, inexact
RR 53-bit reals, inexact, not same as **RDF**
RealField(400) 400-bit reals, inexact
ComplexField(400) complexes, too
ZZ[I] the ring of Gaussian integers
QuadraticField(7) the quadratic field, $\mathbb{Q}(\sqrt{7})$
CyclotomicField(7)
smallest field containing $\mathbb{Q}$ and the zeros of $x^7 - 1$
AA, **QQbar** field of algebraic numbers, $\overline{\mathbb{Q}}$
FiniteField(7) the field $\mathbb{Z}_7$
F.<a> = FiniteField(7^3)
finite field in a of size 7^3 , $\text{GF}(7^3)$
SR ring of symbolic expressions
M.<a>=QQ[sqrt(3)] the field $\mathbb{Q}[\sqrt{3}]$, with $a = \sqrt{3}$.
A.<a,b>=QQ[sqrt(3),sqrt(5)]
the field $\mathbb{Q}[\sqrt{3}, \sqrt{5}]$ with $a = \sqrt{3}$ and $b = \sqrt{5}$.
z = polygen(QQ,'z'); K = NumberField(x^2 - 2,'s')
the number field in s with defining polynomial $x^2 - 2$
s = K.0 set **s** equal to the generator of K
D = ZZ[sqrt(3)]
D.fraction_field()
field of fractions for the integral domain D

Ring Operations

Note: Operations may depend on the ring

A = ZZ[I]; D = ZZ[sqrt(3)] some rings
A.is_ring() is A a ring?
A.is_field() is A a field?
A.is_commutative() is A commutative?
A.is_integral_domain()
True is A an integral domain?
A.is_finite() is A is finite?
A.is_subring(D) is A a subring of D ?
A.order() the number of elements of A
A.characteristic() the characteristic of A
A.zero() the additive identity of A
A.one() the multiplicative identity of A
A.is_exact()
False if A uses a floating point representation

a, b = D.gens(); r = a + b
r.parent() the parent ring of r (in this case, D)
r.is_unit() is r a unit?

Polynomials

R.<x> = ZZ[] R is the polynomial ring $\mathbb{Z}[x]$
R.<x> = QQ[]; R = PolynomialRing(QQ,'x'); **R = QQ['x']**
 R is the polynomial ring $\mathbb{Q}[x]$
S.<z> = Integers(8)[] S is the polynomial ring $\mathbb{Z}_8[z]$
S.<s, t> = QQ[] S is the polynomial ring $\mathbb{Q}[s, t]$
p = 4*x^3 + 8*x^2 - 20*x - 24
a polynomial in R ($= \mathbb{Q}[x]$)
p.is_irreducible() is p irreducible over $\mathbb{Q}[x]$?
q = p.factor() factor p
q.expand() expand q
p.subs(x=3) evaluates p at $x = 3$
R.ideal(p) the ideal in R generated by p
R.cyclotomic_polynomial(7)
the cyclotomic polynomial $x^6 + x^5 + x^4 + x^3 + x^2 + x + 1$
q = x^2-1
p.divides(q) does p divide q ?
p.quo_rem(q)
the quotient and remainder of p upon division by q
gcd(p, q) the greatest common divisor of p and q
p.xgcd(q) the extended gcd of p and q
I = S.ideal([s*t+2,s^3-t^2])
the ideal $(st + 2, s^3 - t^2)$ in S ($= \mathbb{Q}[s, t]$)
S.quotient(I) the quotient ring, S/I

Field Operations

A.<a,b>=QQ[sqrt(3),sqrt(5)]
C.<c> = A.absolute_field()
“flattens” a relative field extension
A.relative_degree()
the degree of the relative extension field
A.absolute_degree()
the degree of the absolute extension
r = a + b; r.minpoly()
the minimal polynomial of the field element r
C.is_galois() is C a Galois extension of Q ?