

Algebraic Number Theory

(PARI-GP version 2.15.4)

Binary Quadratic Forms

create $ax^2 + bxy + cy^2$ **Qfb**(a, b, c) or **Qfb**($[a, b, c]$)
reduce x ($s = \sqrt{D}$, $l = \lfloor s \rfloor$) **qfbred**($x, \{flag\}, \{D\}, \{l\}, \{s\}$)
return $[y, g]$, $g \in \text{SL}_2(\mathbf{Z})$, $y = g \cdot x$ reduced **qfbreds12**(x)
composition of forms $x*y$ or **qfbnucomp**(x, y, l)
 n -th power of form x^n or **qfbnpow**(x, n)
composition **qfbcomp**(x, y)
... without reduction **qfbcomppraw**(x, y)
 n -th power **qfbpow**(x, n)
... without reduction **qfbpowraw**(x, n)
prime form of disc. x above prime p **qfbprimeform**(x, p)
class number of disc. x **qfbclassno**(x)
Hurwitz class number of disc. x **qfbhclassno**(x)
solve $Q(x, y) = n$ in integers **qfbsolve**(Q, n)
solve $x^2 + Dy^2 = p$, p prime **qfbcornacchia**(D, p)
... $x^2 + Dy^2 = 4p$, p prime **qfbcornacchia**($D, 4 * p$)

Quadratic Fields

quadratic number $\omega = \sqrt{x}$ or $(1 + \sqrt{x})/2$ **quadgen**(x)
minimal polynomial of ω **quadpoly**(x)
discriminant of **Q**(\sqrt{x}) **quaddisc**(x)
regulator of real quadratic field **quadregulator**(x)
fundamental unit in O_D , $D > 0$ **quadunit**($D, \{ 'w \}$)
norm of fundamental unit in O_D **quadunitnorm**(D)
index of $O_{Df_2}^\times$ in O_D^\times **quadunitindex**(D, f)
class group of **Q**(\sqrt{D}) **quadclassunit**($D, \{flag\}, \{t\}$)
Hilbert class field of **Q**(\sqrt{D}) **quadhilbert**($D, \{flag\}$)
... using specific class invariant ($D < 0$) **polclass**($D, \{inv\}$)
ray class field modulo f of **Q**(\sqrt{D}) **quadrays**($D, f, \{flag\}$)

General Number Fields: Initializations

The number field $K = \mathbf{Q}[X]/(f)$ is given by irreducible $f \in \mathbf{Q}[X]$.
We denote $\theta = \bar{X}$ the canonical root of f in K . A nf structure contains a maximal order and allows operations on elements and ideals. A bnf adds class group and units. A bnr is attached to ray class groups and class field theory. A rnf is attached to relative extensions L/K .

init number field structure nf **nfinit**($f, \{flag\}$)
 known integer basis B **nfinit**($[f, B]$)
 order maximal at $vp = [p_1, \dots, p_k]$ **nfinit**($[f, vp]$)
 order maximal at all $p \leq P$ **nfinit**($[f, P]$)
 certify maximal order **nfcertify**(nf)

nf members:

a monic $F \in \mathbf{Z}[X]$ defining K **nf.pol**
number of real/complex places **nf.r1/r2/sign**
discriminant of nf **nf.disc**
primes ramified in nf **nf.p**
 T_2 matrix **nf.t2**
complex roots of F **nf.roots**
integral basis of \mathbf{Z}_K as powers of θ **nf.zk**
different/codifferent **nf.diff**, **nf.codiff**
index $[\mathbf{Z}_K : \mathbf{Z}[X]/(F)]$ **nf.index**
recompute nf using current precision **nfnewprec**(nf)
init relative rnf $L = K[Y]/(g)$ **rnfinit**(nf, g)
init bnf structure **bnfinit**($f, 1$)

bnf members: same as nf , plus
 underlying nf **bnf.nf**
 class group, regulator **bnf.clgp**, **bnf.reg**
 fundamental/torsion units **bnf.fu**, **bnf.tu**
 add S -class group and units, yield $bnfS$ **bnfsunit**(bnf, S)
 init class field structure bnr **bnrinit**($bnf, m, \{flag\}$)
bnr members: same as bnf , plus
 underlying bnf **bnr.bnf**
 big ideal structure **bnr.bid**
 modulus m **bnr.mod**
 structure of $(\mathbf{Z}_K/m)^*$ **bnr.zkst**

Fields, subfields, embeddings

Defining polynomials, embeddings
(some) number fields with Galois group G **nflist**(G)
... and $|\text{disc}(K)| = N$ and s complex places **nflist**($G, N, \{s\}$)
... and $a \leq |\text{disc}(K)| \leq b$ **nflist**($G, [a, b], \{s\}$)
smallest poly defining $f = 0$ (slow) **polredabs**($f, \{flag\}$)
small poly defining $f = 0$ (fast) **polredbest**($f, \{flag\}$)
monic integral $g = Cf(x/L)$ **poltomonic**($f, \{\&L\}$)
random Tschirnhausen transform of f **poltschirnhaus**(f)
 $\mathbf{Q}[t]/(f) \subset \mathbf{Q}[t]/(g)$? Isomorphic? **nfisincl**(f, g), **nfisisom**
reverse polmod $a = A(t) \bmod T(t)$ **modreverse**(a)
compositum of $\mathbf{Q}[t]/(f)$, $\mathbf{Q}[t]/(g)$ **polcompositum**($f, g, \{flag\}$)
compositum of $K[t]/(f)$, $K[t]/(g)$ **nfcompositum**($nf, f, g, \{flag\}$)
splitting field of K (degree divides d) **nfsplitting**($nf, \{d\}$)
signs of real embeddings of x **nfeltsign**($nf, x, \{pl\}$)
complex embeddings of x **nfeltembed**($nf, x, \{pl\}$)
 $T \in K[t]$, # of real roots of $\sigma(T) \in R[t]$ **nfpolsturm**($nf, T, \{pl\}$)

Subfields, polynomial factorization

subfields (of degree d) of nf **nfsubfields**($nf, \{d\}$)
maximal subfields of nf **nfsubfieldsmax**(nf)
maximal CM subfield of nf **nfsubfieldscm**(nf)
 $K_d \subset \mathbf{Q}(\zeta_n)$, using Gaussian periods **polsubcyclo**($n, d, \{v\}$)
... using class field theory **polsubcyclofast**(n, d)
roots of unity in nf **nfrootsof1**(nf)
roots of g belonging to nf **nfroots**(nf, g)
factor g in nf **nfactor**(nf, g)

Linear and algebraic relations

poly of degree $\leq k$ with root $x \in \mathbf{C}$ or \mathbf{Q}_p **algdep**(x, k)
alg. dep. with pol. coeffs for series s **seralgdep**(s, x, y)
diff. dep. with pol. coeffs for series s **serdiffdep**(s, x, y)
small linear rel. on coords of vector x **lindep**(x)

Basic Number Field Arithmetic (nf)

Number field elements are **t_INT**, **t_FRAC**, **t_POL**, **t_POLMOD**, or **t_COL**
(on integral basis $nf.zk$).

Basic operations

$x + y$ **nfeltadd**(nf, x, y)
 $x \times y$ **nfeltmul**(nf, x, y)
 x^n , $n \in \mathbf{Z}$ **nfeltpow**(nf, x, n)
 x/y **nfeltdiv**(nf, x, y)
 $q = x \setminus y := \text{round}(x/y)$ **nfeltdiveuc**(nf, x, y)
 $r = x \% y := x - (x \setminus y)y$ **nfeltmod**(nf, x, y)
... $[q, r]$ as above **nfeltdivrem**(nf, x, y)
reduce x modulo ideal A **nfeltreduce**(nf, x, A)
absolute trace $\text{Tr}_{K/\mathbf{Q}}(x)$ **nfelttrace**(nf, x)
absolute norm $N_{K/\mathbf{Q}}(x)$ **nfeltnorm**(nf, x)

is x a square? **nfeltissquare**($nf, x, \{\&y\}$)
... an n -th power? **nfeltispower**($nf, x, n, \{\&y\}$)

Multiplicative structure of K^* ; $K^*/(K^*)^n$
valuation $v_{\mathfrak{p}}(x)$ **nfeltval**(nf, x, \mathfrak{p})
... write $x = \pi^{v_{\mathfrak{p}}(x)}y$ **nfeltval**($nf, x, \mathfrak{p}, \&y$)
quadratic Hilbert symbol (at \mathfrak{p}) **nfhilbert**($nf, a, b, \{\mathfrak{p}\}$)
 b such that $xb^n = v$ is small **idealredmodpower**(nf, x, n)

Maximal order and discriminant

integral basis of field **Q**[x]/(f) **nfbasis**(f)
field discriminant of **Q**[x]/(f) **nfdisc**(f)
... and factorization **nfdiscfactors**(f)
express x on integer basis **nfalgtobasis**(nf, x)
express element x as a polmod **nfbasistoalg**(nf, x)

Hecke Grossencharacters

Let K be a number field and m a modulus. A **gchar** structure describes the group of Hecke Grossencharacters of K of modulus m and allows computations with these characters. A character χ is described by its components modulo $gc.cyc$.

init **gchar** structure gc for modulus m **gcharinit**($bnf, m, \{cm\}$)
gc members:

 underlying bnf **gc.bnf**
 modulus **gc.mod**
 elementary divisors (including 0s) **gc.cyc**
recompute gc using current precision **gcharnewprec**(gc)
evaluate Hecke character chi at ideal id **gchareval**(gc, chi, id)
exponent column of id in \mathbf{R}^n **gcharideallog**(gc, id)
log representation of ideal id **gcharlog**(gc, id)
... of character χ **gcharduallog**(gc, chi)
exponent vector of χ in \mathbf{R}^n **gcharparameters**(gc, chi)
conductor of χ **gcharconductor**(gc, chi)
L-function of χ **lfuncreate**($[gc, chi]$)
local component χ_v of χ **gcharlocal**(gc, chi, v)
 χ s.t. $\chi_v \approx Lchiv[i]$ for $v = Lv[i]$ **gcharidentify**($gc, Lv, Lchiv$)
basis of group of algebraic characters **gcharalgebraic**(gc)
is χ algebraic? **gcharisalgebraic**(gc, chi)

Dedekind Zeta Function ζ_K , Hecke L series

$R = [c, w, h]$ in initialization means we restrict $s \in \mathbf{C}$ to domain $|\Re(s) - c| < w$, $|\Im(s)| < h$; $R = [w, h]$ encodes $[1/2, w, h]$ and $[h]$ encodes $R = [1/2, 0, h]$ (critical line up to height h).

ζ_K as Dirichlet series, $N(I) \leq b$ **dirzetak**(nf, b)
init $\zeta_K^{(k)}(s)$ for $k \leq n$ **L = lfunitinit**($bnf, R, \{n = 0\}$)
compute $\zeta_K(s)$ (n -th derivative) **lfun**($L, s, \{n = 0\}$)
compute $\Lambda_K(s)$ (n -th derivative) **lfunlambda**($L, s, \{n = 0\}$)

init $L_K^{(k)}(s, \chi)$ for $k \leq n$ **L = lfunitinit**($[bnr, chi], R, \{n = 0\}$)
compute $L_K(s, \chi)$ (n -th derivative) **lfun**($L, s, \{n\}$)
Artin root number of K **bnrrootnumber**($bnr, chi, \{flag\}$)
 $L(1, \chi)$, for all χ trivial on H **bnrL1**($bnr, \{H\}, \{flag\}$)

Class Groups & Units (bnf, bnr)

Class field theory data $a_1, \{a_2\}$ is usually bnr (ray class field), bnr, H (congruence subgroup) or bnr, χ (character on **bnr.clgp**). Any of these define a unique abelian extension of K .
units / S -units **bnfunits**($bnf, \{S\}$)
remove GRH assumption from bnf **bnfcertify**(bnf)

expo. of ideal x on class gp	<code>bnfisprincipal(bnf, x, {flag})</code>
... on ray class gp	<code>bnrisprincipal(bnr, x, {flag})</code>
expo. of x on fund. units	<code>bnfisunit(bnf, x)</code>
... on S -units, U is <code>bnfunits(bnf, S)</code>	<code>bnfisunit(bnfs, x, U)</code>
signs of real embeddings of $bnf.fu$	<code>bnfsignunit(bnf)</code>
narrow class group	<code>bnfnarrow(bnf)</code>

Class Field Theory

ray class number for modulus m	<code>bnrclassno(bnf, m)</code>
discriminant of class field	<code>bnrdisc(a1, {a2})</code>
ray class numbers, l list of moduli	<code>bnrclassnolist(bnf, l)</code>
discriminants of class fields	<code>bnrdisclist(bnf, l, {arch}, {flag})</code>
decode output from <code>bnrdisclist</code>	<code>bnfdecodemodule(nf, fa)</code>
is modulus the conductor?	<code>bnrisconductor(a1, {a2})</code>
is class field (bnr, H) Galois over K^G	<code>bnrisgalois(bnr, G, H)</code>
action of automorphism on <code>bnr.gen</code>	<code>bnrgaloismatrix(bnr, aut)</code>
apply <code>bnrgaloismatrix M</code> to H	<code>bnrgaloisapply(bnr, M, H)</code>
characters on <code>bnr.clgp</code> s.t. $\chi(g_i) = e(v_i)$	<code>bnrchar(bnr, g, {v})</code>
conductor of character χ	<code>bnrconductor(bnr, chi)</code>
conductor of extension	<code>bnrconductor(a1, {a2}, {flag})</code>
conductor of extension $K[Y]/(g)$	<code>rnfconductor(bnf, g)</code>
canonical projection $Cl_F \rightarrow Cl_f$, $f \mid F$	<code>bnrmap</code>
Artin group of extension $K[Y]/(g)$	<code>rnfnormgroup(bnr, g)</code>
subgroups of bnr , index $\leq b$	<code>subgrouplist(bnr, b, {flag})</code>
compositum as <code>[bnr, H]</code>	<code>bnrcompositum([bnr1, H1], [bnr2, H2])</code>
class field defined by $H \subset Cl_f$	<code>bnrclassfield(bnr, H)</code>
... low level equivalent, prime degree	<code>rnfkummer(bnr, H)</code>
same, using Stark units (real field)	<code>bnrstark(bnr, sub, {flag})</code>
is a an n -th power in K_v ?	<code>nfislocalpower(nf, v, a, n)</code>
cyclic L/K satisf. local conditions	<code>nfgrunwaldwang(nf, P, D, pl)</code>

Cyclotomic and Abelian fields theory

An Abelian field F given by a subgroup $H \subset (Z/fZ)^*$ is described by an argument F , e.g. f (for $H = 1$, i.e. $Q(\zeta_f)$) or $[G, H]$, where G is `idealstar(f, 1)`, or a minimal polynomial.

minus class number $h^-(F)$	<code>subcyclohminus(F)</code>
... p -part	<code>subcyclohminus(F, p)</code>
minus part of Iwasawa polynomials	<code>subcycloiwasawa(F, p)</code>
p -Sylog of $Cl(F)$	<code>subcyclopclgp(F, p)</code>

Logarithmic class group

logarithmic ℓ -class group	<code>bnflog(bnf, l)</code>
$[\tilde{e}(F_v/Q_p), \tilde{f}(F_v/Q_p)]$	<code>bnflogf(bnf, pr)</code>
$\exp \deg_F(A)$	<code>bnflogdegree(bnf, A, l)</code>
is ℓ -extension L/K locally cyclotomic	<code>rnfislocalcyclo(rnf)</code>

Ideals: elements, primes, or matrix of generators in HNF

is id an ideal in nf ?	<code>nfisideal(nf, id)</code>
is x principal in bnf ?	<code>bnfisprincipal(bnf, x)</code>
give $[a, b]$, s.t. $aZ_K + bZ_K = x$	<code>idealtwoelt(nf, x, {a})</code>
put ideal a ($aZ_K + bZ_K$) in HNF form	<code>idealhnf(nf, a, {b})</code>
norm of ideal x	<code>idealnrm(nf, x)</code>
minimum of ideal x (direction v)	<code>idealmin(nf, x, v)</code>
LLL-reduce the ideal x (direction v)	<code>idealred(nf, x, {v})</code>

Ideal Operations

add ideals x and y	<code>idealadd(nf, x, y)</code>
multiply ideals x and y	<code>idealmul(nf, x, y, {flag})</code>
intersection of ideal x with Q	<code>idealdown(nf, x)</code>
intersection of ideals x and y	<code>idealintersect(nf, x, y, {flag})</code>
n -th power of ideal x	<code>idealpow(nf, x, n, {flag})</code>
inverse of ideal x	<code>idealinv(nf, x)</code>
divide ideal x by y	<code>idealdiv(nf, x, y, {flag})</code>

Algebraic Number Theory

(PARI-GP version 2.15.4)

Find $(a, b) \in x \times y$, $a + b = 1$	<code>idealaddtoone(nf, x, {y})</code>
coprime integral A, B such that $x = A/B$	<code>idealnumden(nf, x)</code>

Primes and Multiplicative Structure

check whether x is a maximal ideal	<code>idealismaximal(nf, x)</code>
factor ideal x in Z_K	<code>idealfactor(nf, x)</code>
expand ideal factorization in K	<code>idealfactorback(nf, f, {e})</code>
is ideal A an n -th power ?	<code>idealispower(nf, A, n)</code>
expand elt factorization in K	<code>nffactorback(nf, f, {e})</code>
decomposition of prime p in Z_K	<code>idealprimedec(nf, p)</code>
valuation of x at prime ideal pr	<code>idealval(nf, x, pr)</code>
weak approximation theorem in nf	<code>idealchinese(nf, x, y)</code>
$a \in K$, s.t. $v_p(a) = v_p(x)$ if $v_p(x) \neq 0$	<code>idealappr(nf, x)</code>
$a \in K$ such that $(a \cdot x, y) = 1$	<code>idealcoprime(nf, x, y)</code>
give bid = structure of $(Z_K/id)^*$	<code>idealstar(nf, id, {flag})</code>
structure of $(1 + \mathfrak{p})/(1 + \mathfrak{p}^k)$	<code>idealprincipalunits(nf, pr, k)</code>
discrete log of x in $(Z_K/bid)^*$	<code>ideallog(nf, x, bid)</code>
idealstar of all ideals of norm $\leq b$	<code>ideallist(nf, b, {flag})</code>
add Archimedean places	<code>ideallistarch(nf, b, {ar}, {flag})</code>
init <code>modpr</code> structure	<code>nfmodprinit(nf, pr, {v})</code>
project t to Z_K/pr	<code>nfmodpr(nf, t, modpr)</code>
lift from Z_K/pr	<code>nfmodprlift(nf, t, modpr)</code>

Galois theory over Q

conjugates of a root θ of nf	<code>nfgaloisconj(nf, {flag})</code>
apply Galois automorphism s to x	<code>nfgaloisapply(nf, s, x)</code>
Galois group of field $Q[x]/(f)$	<code>polgalois(f)</code>
resolvent field of $Q[x]/(f)$	<code>nfresolvent(f)</code>
initializes a Galois group structure G	<code>galoisinit(pol, {den})</code>
... for the splitting field of pol	<code>galoisplittinginit(pol, {d})</code>
character table of G	<code>galoischartable(G)</code>
conjugacy classes of G	<code>galoisconjclasses(G)</code>
$\det(1 - \rho(g)T)$, χ character of ρ	<code>galoischarpoly(G, x, {o})</code>
$\det(\rho(g))$, χ character of ρ	<code>galoischarhet(G, x, {o})</code>
action of p in <code>nfgaloisconj</code> form	<code>galoispermtpol(G, {p})</code>
identify as abstract group	<code>galoisidentify(G)</code>
export a group for GAP/MAGMA	<code>galoisexport(G, {flag})</code>
subgroups of the Galois group G	<code>galoissubgroups(G)</code>
is subgroup H normal?	<code>galoisisnormal(G, H)</code>
subfields from subgroups	<code>galoissubfields(G, {flag}, {v})</code>
fixed field	<code>galoisfixedfield(G, perm, {flag}, {v})</code>
Frobenius at maximal ideal P	<code>idealfrobenius(nf, G, P)</code>
ramification groups at P	<code>idealramgroups(nf, G, P)</code>
is G abelian?	<code>galoisisabelian(G, {flag})</code>
abelian number fields/ Q	<code>galoissubcyclo(N, H, {flag}, {v})</code>

The galpol package

query the package: polynomial	<code>galoisgetpol(a, b, {s})</code>
...: permutation group	<code>galoisgetgroup(a, b)</code>
...: group description	<code>galoisgetname(a, b)</code>

Relative Number Fields (rnf)

Extension L/K is defined by $T \in K[x]$.

absolute equation of L	<code>rnfequation(nf, T, {flag})</code>
is L/K abelian?	<code>rnfisabelian(nf, T)</code>
relative <code>nfalgtobasis</code>	<code>rnfalgtobasis(rnf, x)</code>
relative <code>nfbasistoalg</code>	<code>rnfbasistoalg(rnf, x)</code>
relative <code>idealhnf</code>	<code>rnfidealhnf(rnf, x)</code>
relative <code>idealmul</code>	<code>rnfidealmul(rnf, x, y)</code>
relative <code>idealtwoelt</code>	<code>rnfidealtwoelt(rnf, x)</code>

Lifts and Push-downs

absolute \rightarrow relative representation for x	<code>rnfeltabstorel(rnf, x)</code>
relative \rightarrow absolute representation for x	<code>rnfeltretloabs(rnf, x)</code>
lift x to the relative field	<code>rnfeltup(rnf, x)</code>
push x down to the base field	<code>rnfeltdown(rnf, x)</code>
idem for x ideal: <code>(rnfideal)reltoabs, abstorel, up, down</code>	

Norms and Trace

relative norm of element $x \in L$	<code>rnfeltnrm(rnf, x)</code>
relative trace of element $x \in L$	<code>rnfelttrace(rnf, x)</code>
absolute norm of ideal x	<code>rnfidealnrmabs(rnf, x)</code>
relative norm of ideal x	<code>rnfidealnrmrel(rnf, x)</code>
solutions of $N_{K/Q}(y) = x \in Z$	<code>bnfisintnrm(bnf, x)</code>
is $x \in Q$ a norm from K ?	<code>bnfisnorm(bnf, x, {flag})</code>
initialize T for norm eq. solver	<code>rnfisnorminit(K, pol, {flag})</code>
is $a \in K$ a norm from L ?	<code>rnfisnorm(T, a, {flag})</code>
initialize t for Thue equation solver	<code>thueinit(f)</code>
solve Thue equation $f(x, y) = a$	<code>thue(t, a, {sol})</code>
characteristic poly. of $a \bmod T$	<code>rnfcharpoly(nf, T, a, {v})</code>

Factorization

factor ideal x in L	<code>rnfidealfactor(rnf, x)</code>
$[S, T]: T_{i,j} \mid S_i$; S primes of K above p	<code>rnfidealprimedec(rnf, p)</code>

Maximal order Z_L as a Z_K -module

relative <code>polredbest</code>	<code>rnfpolredbest(nf, T)</code>
relative <code>polredabs</code>	<code>rnfpolredabs(nf, T)</code>
relative Dedekind criterion, prime pr	<code>rnfdedekind(nf, T, pr)</code>
discriminant of relative extension	<code>rnfdisc(nf, T)</code>
pseudo-basis of Z_L	<code>rnfpseudobasis(nf, T)</code>

General Z_K -modules: $M = [\text{matrix, vec. of ideals}] \subset L$

relative HNF / SNF	<code>nfhnf(nf, M), nfsnf</code>
multiple of $\det M$	<code>nfdetint(nf, M)</code>
HNF of M where $d = nfdetint(M)$	<code>nfhnfmod(x, d)</code>
reduced basis for M	<code>rnfilllgram(nf, T, M)</code>
determinant of pseudo-matrix M	<code>rnfdet(nf, M)</code>
Steinitz class of M	<code>rnfstteinitz(nf, M)</code>
Z_K -basis of M if Z_K -free, or 0	<code>rnfhnfbasis(bnf, M)</code>
n -basis of M , or $(n + 1)$ -generating set	<code>rnfbasis(bnf, M)</code>
is M a free Z_K -module?	<code>rnfisfree(bnf, M)</code>

Associative Algebras

A is a general associative algebra given by a multiplication table *mt* (over **Q** or **F_p**); represented by *al* from `algtblinit`.
create *al* from *mt* (over **F_p**) `algtblinit(mt, {p = 0})`
group algebra **Q**[*G*] (or **F_p**[*G*]) `alggroup(G, {p = 0})`
center of group algebra `alggrouppcenter(G, {p = 0})`
Properties
is (*mt*, *p*) OK for `algtblinit`? `algisassociative(mt, {p = 0})`
multiplication table *mt* `algmultable(al)`
dimension of *A* over prime subfield `algdim(al)`
characteristic of *A* `algchar(al)`
is *A* commutative? `algiscommutative(al)`
is *A* simple? `algissimple(al)`
is *A* semi-simple? `algissemisimple(al)`
center of *A* `algcenter(al)`
Jacobson radical of *A* `algradical(al)`
radical *J* and simple factors of *A*/*J* `algsimpledec(al)`
Operations on algebras
create *A*/*I*, *I* two-sided ideal `algquotient(al, I)`
create *A*₁ ⊗ *A*₂ `algtensor(al1, al2)`
create subalgebra from basis *B* `algsubalg(al, B)`
quotients by ortho. central idempotents *e* `algcentralproj(al, e)`
isomorphic alg. with integral mult. table `algmakeintegral(mt)`
prime subalgebra of semi-simple *A* over **F_p** `algprimesubalg(al)`
find isomorphism *A* ≅ *M_d*(**F_q**) `algsplit(al)`
Operations on lattices in algebras
lattice generated by cols. of *M* `alglathnf(al, M)`
... by the products *xy*, *x* ∈ *lat1*, *y* ∈ *lat2* `alglatmul(al, lat1, lat2)`
sum *lat1* + *lat2* of the lattices `alglatadd(al, lat1, lat2)`
intersection *lat1* ∩ *lat2* `alglatinter(al, lat1, lat2)`
test *lat1* ⊂ *lat2* `alglatsubset(al, lat1, lat2)`
generalized index (*lat2* : *lat1*) `alglatindex(al, lat1, lat2)`
{*x* ∈ *al* | *x* · *lat1* ⊂ *lat2*} `alglatlefttransporter(al, lat1, lat2)`
{*x* ∈ *al* | *lat1* · *x* ⊂ *lat2*} `alglatrighttransporter(al, lat1, lat2)`
test *x* ∈ *lat* (set *c* = coord. of *x*) `alglatcontains(al, lat, x, {&c})`
element of *lat* with coordinates *c* `alglatelement(al, lat, c)`
Operations on elements
a + *b*, *a* − *b*, −*a* `algadd(al, a, b), algsub, algneg`
a × *b*, *a*² `algmul(al, a, b), algsqr`
aⁿ, *a*^{−1} `algpow(al, a, n), alginv`
is *x* invertible ? (then set *z* = *x*^{−1}) `alginv(al, x, {&z})`
find *z* such that *x* × *z* = *y* `algdivl(al, x, y)`
find *z* such that *z* × *x* = *y* `algdivr(al, x, y)`
does *z* s.t. *x* × *z* = *y* exist? (set it) `algisdivl(al, x, y, {&z})`
matrix of *v* ↦ *x* · *v* `algtomatrix(al, x)`
absolute norm `algnorm(al, x)`
absolute trace `algtrace(al, x)`
absolute char. polynomial `algcharpoly(al, x)`
given *a* ∈ *A* and polynomial *T*, return *T*(*a*) `algpoleval(al, T, a)`
random element in a box `algrandom(al, b)`

Central Simple Algebras

A is a central simple algebra over a number field *K*; represented by *al* from `alginitt`; *K* is given by a *nf* structure.
create CSA from data `alginitt(B, C, {v}, {maxord = 1})`
multiplication table over *K* *B* = *K*, *C* = *mt*
cyclic algebra (*L*/*K*, *σ*, *b*) *B* = *rnf*, *C* = [*sigma*, *b*]
quaternion algebra (*a*, *b*)_{*K*} *B* = *K*, *C* = [*a*, *b*]
matrix algebra *M_d*(*K*) *B* = *K*, *C* = *d*
local Hasse invariants over *K* *B* = *K*, *C* = [*d*, [*PR*, *HF*], *HI*]

Properties

type of *al* (*mt*, CSA) `algtype(al)`
dimension of *A* over **Q** `algdim(al, 1)`
dimension of *al* over its center *K* `algdim(al)`
degree of *A* (= √dim_{*K*} *A*) `algdegree(al)`
al a cyclic algebra (*L*/*K*, *σ*, *b*); return *σ* `algaut(al)`
...return *b* `algb(al)`
...return *L*/*K*, as an *rnf* `algsplittingfield(al)`
split *A* over an extension of *K* `algsplittingdata(al)`
splitting field of *A* as an *rnf* over center `algsplittingfield(al)`
multiplication table over center `algrelmultable(al)`
places of *K* at which *A* ramifies `algramifiedplaces(al)`
Hasse invariants at finite places of *K* `alghassef(al)`
Hasse invariants at infinite places of *K* `alghassei(al)`
Hasse invariant at place *v* `alghasse(al, v)`
index of *A* over *K* (at place *v*) `algindex(al, {v})`
is *al* a division algebra? (at place *v*) `algisdivision(al, {v})`
is *A* ramified? (at place *v*) `algisramified(al, {v})`
is *A* split? (at place *v*) `algisplit(al, {v})`

Operations on elements

reduced norm `algnorm(al, x)`
reduced trace `algtrace(al, x)`
reduced char. polynomial `algcharpoly(al, x)`
express *x* on integral basis `algalgtobasis(al, x)`
convert *x* to algebraic form `algbasistoalg(al, x)`
map *x* ∈ *A* to *M_d*(*L*), *L* split. field `algtomatrix(al, x)`

Orders

Z-basis of order *O*₀ `algbasis(al)`
discriminant of order *O*₀ `algdisc(al)`
Z-basis of natural order in terms *O*₀'s basis `alginvbasis(al)`