

# Algebraic Number Theory

(PARI-GP version 2.15.5)

## 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  $\omega$  **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**( $\sqrt{D}$ ) **quadray**( $D, f, \{flag\}$ )

## 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  $\theta$  **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**( $f$ )  
**Q**[ $t$ ]/( $f$ )  $\subset$  **Q**[ $t$ ]/( $g$ ) ? Isomorphic? **nfisincl**( $f, g$ ), **nfisisom**  
reverse polmod  $a = A(t) \bmod T(t)$  **modreverse**( $a$ )  
compositum of **Q**[ $t$ ]/( $f$ ), **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*,  $x, y$ )  
 $q = x \backslash y := \text{round}(x/y)$  **nfeltdivideuc**(*nf*,  $x, y$ )  
 $r = x \backslash y := x - (x \backslash y)y$  **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  $\chi$  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  $\chi$  **gcharduallog**(*gc*, *chi*)  
exponent vector of  $\chi$  in  $\mathbf{R}^n$  **gcharparameters**(*gc*, *chi*)  
conductor of  $\chi$  **gcharconductor**(*gc*, *chi*)  
L-function of  $\chi$  **lfuncreate**(*gc*, *chi*)  
local component  $\chi_v$  of  $\chi$  **gcharlocal**(*gc*, *chi*,  $v$ )  
 $\chi$  s.t.  $\chi_v \approx Lchiv[i]$  for  $v = Lv[i]$  **gcharidentify**(*gc*,  $Lv$ , *Lchiv*)  
basis of group of algebraic characters **gcharalgebraic**(*gc*)  
is  $\chi$  algebraic? **gcharisalgebraic**(*gc*, *chi*)

### Dedekind Zeta Function $\zeta_K$ , 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  $h$ ).

$\zeta_K$  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  $\chi$  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*,  $\chi$  (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      `bnfisprincipal(bnf,x,{flag})`  
...on ray class gp      `bnrisprincipal(bnr,x,{flag})`  
expo. of  $x$  on fund. units      `bnfisunit(bnf,x)`  
...on  $S$ -units,  $U$  is `bnfunits(bnf,S)`      `bnfisunit(bnfs,x,U)`  
signs of real embeddings of  $bnf$ .fu      `bnfsignunit(bnf)`  
narrow class group      `bnfnarrow(bnf)`

**Class Field Theory**

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

**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  $h^-(F)$       `subcyclohminus(F)`  
... $p$ -part      `subcyclohminus(F,p)`  
minus part of Iwasawa polynomials      `subcycloiwasawa(F,p)`  
 $p$ -Sylow of  $\text{Cl}(F)$       `subcyclopclgp(F,p)`

**Logarithmic class group**

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

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

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

**Ideal Operations**

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

**Algebraic Number Theory**

(PARI-GP version 2.15.5)

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

**Primes and Multiplicative Structure**

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

**Galois theory over Q**

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

**The galpol package**

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

**Relative Number Fields (rnf)**

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

absolute equation of  $L$       `rnfequation(nf,T,{flag})`  
is  $L/K$  abelian?      `rnfisabelian(nf,T)`  
relative `nfalttobasis`      `rnfalttobasis(rnf,x)`  
relative `nfbasistoalg`      `rnfbasistoalg(rnf,x)`  
relative `idealhnf`      `rnfidealhnf(rnf,x)`  
relative `idealmul`      `rnfidealmul(rnf,x,y)`  
relative `idealtwoelt`      `rnfidealtwoelt(rnf,x)`

**Lifts and Push-downs**

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

**Norms and Trace**

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

**Factorization**

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

**Maximal order  $\mathbf{Z}_L$  as a  $\mathbf{Z}_K$ -module**

relative `polredbest`      `rnfpolredbest(nf,T)`  
relative `polredabs`      `rnfpolredabs(nf,T)`  
relative Dedekind criterion, prime  $pr$       `rnfdedekind(nf,T,pr)`  
discriminant of relative extension      `rnfdisc(nf,T)`  
pseudo-basis of  $\mathbf{Z}_L$       `rnfpseudobasis(nf,T)`

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

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

Associative Algebras

*A* is a general associative algebra given by a multiplication table *mt* (over **Q** or **F<sub>p</sub>**); represented by *al* from `algtableinit`.  
create *al* from *mt* (over **F<sub>p</sub>**)                    `algtableinit(mt, {p = 0})`  
group algebra **Q**[*G*] (or **F<sub>p</sub>**[*G*])                    `alggroup(G, {p = 0})`  
center of group algebra                    `alggrouppcenter(G, {p = 0})`  
**Properties**  
is (*mt*, *p*) OK for `algtableinit`?                    `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*<sub>1</sub> ⊗ *A*<sub>2</sub>                    `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<sub>p</sub>**                    `algprimesubalg(al)`  
find isomorphism *A* ≅ *M<sub>d</sub>*(**F<sub>q</sub>**)                    `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*<sup>2</sup>                    `algmul(al, a, b), algsqr`  
*a<sup>n</sup>*, *a*<sup>−1</sup>                    `algpow(al, a, n), alginv`  
is *x* invertible ? (then set *z* = *x*<sup>−1</sup>)                    `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)                    `algsdivl(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 `algininit`; *K* is given by a *nf* structure.  
create CSA from data                    `algininit(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*)<sub>*K*</sub>                    *B* = *K*, *C* = [*a*, *b*]  
matrix algebra *M<sub>d</sub>*(*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<sub>*K*</sub> *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*)                    `algsdivision(al, {v})`  
is *A* ramified? (at place *v*)                    `algsramified(al, {v})`  
is *A* split? (at place *v*)                    `algsisplit(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<sub>d</sub>*(*L*), *L* split. field                    `algtomatrix(al, x)`

Orders

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