Some new GP features

A tutorial

B. Allombert

IMB
CNRS/Université de Bordeaux

06/01/2025

factorlimit

We have changed the way the table of prime numbers is stored:

- The default `primelimit` is 2^{20} instead of 510^5.
- Primes are now indexed, so functions like prime, primepi etc are much faster.

```
? sum(i=1,5*10^5,primepi(i))
%1 = 10854805343
? ##
  ***   last result computed in 41 ms.
```

- a new default `factorlimit` (2^{20} by default) is used for fast trial factorization using a sieve.

```
? for(i=1,100,factor(3^1000-1,5*10^5))
? ##
  ***   last result computed in 90 ms.
```

mapapply

`mapapply( M,x,f,u)` if *x* is in the domain of *M*, apply *f* to the image of *x*, otherwise returns *u*(). For example, maplistput allow to append *v* to the value of *k*:

```
? maplistput(~M, k, v) =
{
  mapapply(~M, k,
       (~y)->listput(~y,v),
       ()->List(v));
}
```

mapapply

```
? M = Map();
%6 = Map([;])
? maplistput(~M, "a", 1); M
%7 = Map(["a", List([1])])
? maplistput(~M, "a", 2); M
%8 = Map(["a", List([1, 2])])
? maplistput(~M, "b", 3); M
%9 = Map(["a", List([1, 2]); "b", List([3])])
? maplistput(~M, "a", 4); M
%10 = Map(["a", List([1, 2, 4]); "b", List([])])
```

nfweilheight

`nfweilheight(nf, v)` returns the absolute logarithmic Weil height of *v* seen as an element of the projective space over `nf`.

```
? Q = nfinit(x); Qi = nfinit(x^2 + 1);
? nfweilheight(Q, [1, 2, -3, 101])
%12 = 4.6151205168412594508841982669129891569
? nfweilheight(Qi, [1, 2, -3, 101])
%13 = 4.6151205168412594508841982669129891569
? nfweilheight(Q, [1, x+2, x+3])
%14 = 1.0986122886681096913952452369225257047
```

bnrstarkunit

Returns the characteristic polynomial of the (conjectural) Stark unit corresponding to class field theory parameters. this define an abelian extens The ground field attached to bnr must be totally real and all but one infinite place must become complex in the class field, which must be a quadratic extension of its totally real subfield.

```
? bnf = bnfinit(y^2 - 2);
? bnr = bnrinit(bnf, [15, [1,0]]);
? P = lift(bnrstarkunit(bnr))
%17 = x^8 + (-9000*y - 12728)*x^7 + (57877380*y + 8
? rnfisabelian(bnf,P)
%18 = 1
? rnfconductor(bnf, P)[1]
%19 = [[15,0;0,15],[1,0]]
```

ellmaninconstant

`ellmaninconstant(E)` return the Manin constant of the rational elliptic curve E.

```
? ellmaninconstant(ellinit("11a1"))
%1 = 1
? ellmaninconstant(ellinit("11a2"))
%2 = 1
? ellmaninconstant(ellinit("11a3"))
%3 = 5
```

ellchangecompose, ellchangeinvert

```
? E=ellinit([1,2,3,4,5]);
? F=ellinit([61/16,127/32]);
? urst=ellisisom(E,F)
%23 = [1,-3/4,-1/2,-9/8]
? ellchangecurve(E,urst)[1..5]
%24 = [0,0,0,61/16,127/32]
? ursti = ellchangeinvert(E,urst)
%25 = [1,3/4,1/2,3/2]
? ellchangecurve(F,ursti)[1..5]
%26 = [1,2,3,4,5]
? ellchangecompose(E,urst,ursti)
%27 = [1,0,0,0]
```

hyperellordinate

hyperellordinate(H,x): y-coordinates corresponding to x-ordinate x on hyperelliptic curve H.

```
? P = x^6+3*x+2;
? hyperellordinate(P,1/2)
%29 = [15/8,-15/8]
? hyperellisoncurve(P,[1/2,15/8])
%30 = 1
? W = [x^6+5, x];
? hyperellordinate(W,1)
%32 = [2,-3]
? hyperellisoncurve(W,[1,2])
%33 = 1
```

derivative of `psi`

It is now possible to compute the derivatives of `psi`.

```
? (psi(21,10)-psi(1,10))/10!
%34 = 1.0004941886041119179811698149249862114
? harmonic(20,11)*1.
%35 = 1.0004941886041119179811698149249862114
? (psi(21+O(101^10),10)-psi(1+O(101^10),10))/10!
%36 = 77+32*101+50*101^2+8*101^3+74*101^4+81*101^5+
? harmonic(20,11)+O(101^10)
%37 = 77+32*101+50*101^2+8*101^3+74*101^4+81*101^5+
```

p-adic Hurwitz zeta with character

When *s* is p-adic, the third parameter of zetahurwitz is the power of the Teichmuller character to use.

```
? zetahurwitz(3+O(11^7),0,0)
%38 = 5*11^-1+1+10*11+9*11^2+5*11^3+8*11^4+3*11^5+O
? zeta(3+O(11^7))
%39 = 5*11^-1+1+10*11+9*11^2+5*11^3+8*11^4+3*11^5+O
? zetahurwitz(3+O(11^7),0,-2)
%40 = 5+4*11+11^2+5*11^4+8*11^5+O(11^6)
? -psi(O(11^7),2)/2!
%41 = 5+4*11+11^2+5*11^4+8*11^5+O(11^6)
```

Algebras

new GP functions algnewprec, algeltfromnf, algmodprinit, algmodpr, algmodprlift, algeichlerbasis, algquattobasis, algbasistoquat, algisquatalg