MagmaTutorial

May 4, 2023

0.1 Land acknowledgement

I respectfully acknowledge we are presently on the unceded traditional territories of the Coast Salish peoples, including the səlilwətaɬ (Tsleil-Waututh), k ik ə əm (Kwikwetlem), Sḵwx̱wú7mesh Úxwumixw (Squamish) and x mə k əyəm (Musqueam) Nations.

1 Introduction to Magma

Basic model for most computer algebra systems: *glorified graphing calculator*. They tend to be built around a read-evaluate-print-loop (REPL)" - you type in instructions that the system **reads** - the system **evaluates** (executes) the instructions - the system **prints** the result (if appropriate)

The instructions may also change state of the system (storing things in memory etc.). Using it looks something like this: Python and many interactive programming languages work in the same way.

The interface you see here is a *Jupyter notebook*. It offers a convenient way to communicate with a REPL when you are developing or experimenting.

[1]:
```
a:=10;a;
```

```
10
```

[2]:
```
for i in [1..10] do
    a := a + 1;
    print(a);
end for;
```

```
11
12
13
14
15
16
17
18
19
20
```

1.1 Some basic computations in Magma

How about factoring a polynomial:

[3]:
```
R<x>:=PolynomialRing(Rationals());
Factorization(x^4-1);
```

```
[
<x - 1, 1>,
<x + 1, 1>,
<x^2 + 1, 1>
]
```

[4]:
```
F<a>:=GF(64);
f:=MinimalPolynomial(a);
```

[5]:
```
rr<y>:=Parent(f);
rr;
f;
```

```
Univariate Polynomial Ring in y over GF(2)
y^6 + y^4 + y^3 + y + 1
```

1.2 Observations:

- Magma uses `<` and `>` as a special kind of bracket (as a result, equality and ordering relations are spelled `eq`, `ne`, `lt`, `le`, `gt`, `le`)
- In order to define a polynomial, one first defines a polynomial ring and gives a name to the generator.
- One can then create the polynomials. Because Magma is explicit about the ring a polynomial lies in, questions about factorizations are unambiguous.

1.3 Some algebraic number theory

As an example where Magma's sometimes seemingly pedantic insistence on explicitly defined mathematical context starts to pay of, let's look at some basic algebraic number theory. We consider a finite extension of $\mathbb{Q}$, i.e., a *number field*:

$$K = \mathbb{Q}(a) = \mathbb{Q}[x]/(x^5 + x + 15).$$

We determine the subring $\mathcal{O}_K$ of elements that are integral over $\mathbb{Z}$. This is called the *ring of integers* of K.

The ring $\mathcal{O}_K$ is a Dedekind domain. Dirichlet's Unit Theorem tells us that $\mathcal{O}_K$

, and hence it has unique ideal factorization. The only obstruction to it being a unique factorization domain itself is that it might not be a Principal Ideal Domain.

[6]:
```
R<x>:=PolynomialRing(Rationals());
K<a>:=NumberField(x^5+x+15);
K;
```

```
a^5;
```

```
Number Field with defining polynomial x^5 + x + 15 over the Rational Field
-a - 15
```

[7]:
```
G,l,data:=GaloisGroup(K);
Parent(l);
```

```
Set of sequences over Unramified extension defined by the polynomial (1 +
O(41^20))*x^2 + (38 + O(41^20))*x + 6 + O(41^20) over 41-adic ring
```

[8]:
```
OK:=IntegerRing(K);OK;
```

```
Maximal Equation Order with defining polynomial x^5 + x + 15 over its ground
order
```

Dirichlet's Unit Theorem tells us that the group of units $\mathcal{O}_K^\times$ is finitely generated. Magma can compute it:

[9]:
```
U,m:=UnitGroup(OK);
U;
print "---";
m;
```

```
Abelian Group isomorphic to Z/2 + Z + Z
Defined on 3 generators
Relations:
2*U.1 = 0
---
Mapping from: GrpAb: U to RngOrd: OK
```

Note the representation of the unit group: It is returned as an abstract finitely generated abelian group, together with a mapping from the group into $\mathcal{O}_K$. The mapping allows us to go back and forth between the two.

[10]:
```
U.1, U.2, U.3;
```

```
U.1
U.2
U.3
```

[11]:
```
[K!m(U.i): i in [1,2,3]];
```

```
[
-1,
505*a^4 - 540*a^3 - 864*a^2 + 2549*a + 151,
4*a^4 - 3*a^3 - 7*a^2 + 18*a + 4
]
```

[12]:
```
1/(K!m(U.3));
```

```
3589*a^4 - 6024*a^3 + 10111*a^2 - 16971*a + 32074
```

We can check that the generators of the unit group are indeed units by verifying that their minimal polynomials have a unit constant term in $\mathbb{Z}$.

[13]:
```
f:=MinimalPolynomial(m(U.2));
f;
```

```
$.1^5 + 1265*$.1^4 + 128060607*$.1^3 - 4382382*$.1^2 + 1330*$.1 - 1
```

Note the odd `$.1`. That is how Magma prints generators it doesn't know a name for. To fix it, we should give a name to the generator, which we can obtain from the `Parent`. If you use Magma a bit you will encounter these `$.1` quite a bit, so it's good to learn how to remedy them.

[14]:
```
Zx<xZ>:=Parent(f);
```

The ring $\mathcal{O}_K$ is a Dedekind domain. That means the only obstruction to it being a unique factorization domain is that it may fail to be a principal ideal domain. The *Ideal Class Group* measures the index of the subgroup generated by principal ideals in the group of (fractional) ideals of $\mathcal{O}_K$. It is a finite group and Magma can compute it:

[15]:
```
P3s:=Factorization(3*OK);P3s;
```

```
[
<Prime Ideal of OK
Two element generators:
[3, 0, 0, 0, 0]
[0, 1, 0, 0, 0], 1>,
<Prime Ideal of OK
Two element generators:
[3, 0, 0, 0, 0]
[5, 1, 1, 0, 0], 1>,
<Prime Ideal of OK
Two element generators:
[3, 0, 0, 0, 0]
[2, 2, 1, 0, 0], 1>
]
```

[16]:
```
Pa:=P3s[3][1];
Parent(Pa);
```

```
Set of ideals of Maximal Equation Order with defining polynomial x^5 + x + 15
over its ground order
```

[17]:
```
bl,gen:=IsPrincipal(Pa);MinimalPolynomial(gen);
```

```
xZ^5 + 77*xZ^4 + 2713*xZ^3 + 1808*xZ^2 + 336*xZ + 9
```

[18]:
```
[IsPrincipal(p[1]) : p in P3s];
```

```
[ false, false, true ]
```

[19]:
```
Cl, mp:=ClassGroup(OK);
Cl;
print "---";
mp;
```

```
Abelian Group isomorphic to Z/2
Defined on 1 generator
Relations:
2*Cl.1 = 0
---
Mapping from: GrpAb: Cl to Set of ideals of OK
```

[20]:
```
(Cl.1)@mp;
```

```
Ideal of OK
Two element generators:
[2, 0, 0, 0, 0]
[1, 1, 1, 0, 0]
```

1.4 Evaluating a mapping

Magma has its roots in the group theory community, where actions are often on the right. Mapping applications in Magma are even originally on the right: `f(a)` is actually just a different notation for `a@f`. As a result `f(g(a))` the same as `a@g@f`, which is the same as `a@(g*f)` (where `*` is function composition), so function composition `g*f` in magma actually read as "`g` then `f`". So `(g*f)(a)` is not what you might think it is. This is a good gotcha to be aware of.

Some mappings may have an "inverse" or at least a way to select a preimage. This can be accessed by `a@@f`.

[21]:
```
[p[1]@@mp : p in P3s];
```

```
[
Cl.1,
Cl.1,
0
]
```

1.5 Magma's Documentation

Magma has quite extensive documentation, but it takes a little effort to find your way around in it:

http://magma.maths.usyd.edu.au/magma/documentation/

I recommend that at some point you read First Steps in Magma. It describes the basic principles of Magma.

The Magma Handbook is the main reference. You can use "Search" to find topics, but do read the introduction of the relevant section. You should probably read the chapter on (The Magma Language)[http://magma.maths.usyd.edu.au/magma/handbook/part/1] in its entirety at some point.

[22]:
```
?Sequence
```

```
Link to http://magma.maths.usyd.edu.au/magma/handbook/search?
  ↪chapters=1&examples=1&intrinsics=1&query=Sequence
```

[23]:
```
PolynomialRing;
```

```
Intrinsic 'PolynomialRing'

Signatures:

(LS::LinearSys) -> RngMPol

The polynomial ring of LS.

(model::ModelG1) -> RngMPol

The polynomial ring used to define the given genus one model.

(R::Rng) -> RngUPol
[
Global: BoolElt,
Exact: BoolElt
]

Create the univariate polynomial ring over R.

(R::Rng, n::RngIntElt) -> RngMPol
[
Global: BoolElt,
Sparse: BoolElt
]

Create a multivariate polynomial ring over R in n variables.

(R::Rng, n::RngIntElt, O::MonStgElt) -> RngMPol
(R::Rng, n::RngIntElt, O::MonStgElt, x1::Any) -> RngMPol
(R::Rng, n::RngIntElt, O::MonStgElt, x1::Any, x2::Any) -> RngMPol
(R::Rng, n::RngIntElt, O::MonStgElt, x1::Any, x2::Any, x3::Any) -> RngMPol

Create the multivariate polynomial ring over R in n variables with given order
O.

(R::Rng, n::RngIntElt, T::Tup) -> RngMPol

Create the multivariate polynomial ring over R in n variables with the order
described by tuple T (matching what is returned by MonomialOrder).
```

```
(R::Rng, G::SeqEnum[RngIntElt]) -> RngMPol

Create the graded multivariate polynomial ring over R in #G variables and with
the given grading G on the variables.

(R::Rng, G::SeqEnum[RngIntElt], O::MonStgElt) -> RngMPol
(R::Rng, G::SeqEnum[RngIntElt], O::MonStgElt, x1::Any) -> RngMPol
(R::Rng, G::SeqEnum[RngIntElt], O::MonStgElt, x1::Any, x2::Any) -> RngMPol

Create the graded multivariate polynomial ring over R in #G variables with the
given grading G on the variables and with the given order O.

(R::Rng, G::SeqEnum[RngIntElt], T::Tup) -> RngMPol

Create the graded multivariate polynomial ring over R in #G variables with the
given grading G on the variables and with the order described by tuple T
(matching what is returned by MonomialOrder).

(R::RngInvar) -> RngMPol

The generic polynomial ring in which the elements of R lie.
```

1.6 A more elaborate example

There is a classical algebraic-geometric construction that takes 7 points in the projective plane and constructs a plane quartic curve from it. For the algebraic geometers in the audience: blowing up 7 points in $\mathbb{P}^2$ (in general position) yields a degree-2 del Pezzo surface, for which the anti-canonical system yields a double cover of $\mathbb{P}^2$, branched over a quartic curve.

The construction is the following: - We determine the space of cubics that vanish at the 7 points. That space is 3-dimensional; with basis, say, f_1, f_2, f_3 - These define a map $\phi\colon : \mathbb{P}^2 \to \mathbb{P}^2$ defined by $(f_1 : f_2 : f_3)$ - The map ϕ has a 3×3 Jacobian matrix associated with it of the partial derivatives of f_1, f_2, f_3. Its determinant defined a sextic curve D in $\mathbb{P}^2$, passing through the seven points. - The image $C = \phi(D)$ is a quartic plane curve. The map ϕ actually defines a birational map between C and D.

[24]:
```
P:=[[1,0,0],[0,1,0],[0,0,1],[1,1,1],[1,3,-1],[1,2,7],[1,-1,-2]];
P;
```

```
[
[ 1, 0, 0 ],
[ 0, 1, 0 ],
[ 0, 0, 1 ],
[ 1, 1, 1 ],
[ 1, 3, -1 ],
[ 1, 2, 7 ],
[ 1, -1, -2 ]
```

```
]
```

[25]:
```
P2uvw<u,v,w>:=ProjectiveSpace(Rationals(),2);
```

[26]:
```
[P2uvw!p : p in P];
```

```
[ (1 : 0 : 0), (0 : 1 : 0), (0 : 0 : 1), (1 : 1 : 1), (-1 : -3 : 1), (1/7 : 2/7
: 1), (-1/2 : 1/2 : 1) ]
```

[27]:
```
mons:=MonomialsOfDegree(Parent(w),3);mons;
```

```
{@
u^3,
u^2*v,
u^2*w,
u*v^2,
u*v*w,
u*w^2,
v^3,
v^2*w,
v*w^2,
w^3
@}
```

[28]:
```
K:=Kernel(Matrix([[Evaluate(m,p): p in P]:m in mons]));
```

[29]:
```
PHI:=[&+[b[i]*mons[i]: i in [1..#mons]]: b in Basis(K)];
```

[30]:
```
P2xyz<x,y,z>:=ProjectiveSpace(Rationals(),2);
phi:=map<P2uvw->P2xyz|PHI>;phi;
```

```
Mapping from: Prj: P2uvw to Prj: P2xyz
with equations :
u^2*v - 208/63*u*v*w + 19/14*v^2*w + 37/21*u*w^2 - 103/126*v*w^2
u^2*w - 67/27*u*v*w + 2/3*v^2*w + 13/9*u*w^2 - 17/27*v*w^2
u*v^2 - 85/21*u*v*w + 31/14*v^2*w + 13/7*u*w^2 - 43/42*v*w^2
```

[31]:
```
Support(BaseScheme(phi));
```

```
{ (0 : 1 : 0), (1 : 0 : 0), (1 : 1 : 1), (0 : 0 : 1), (1/7 : 2/7 : 1), (-1/2 :
1/2 : 1), (-1 : -3 : 1) }
```

[32]:
```
jacobian_matrix:=Matrix([[Derivative(PHI[i],j): i in [1..3]]: j in [1..3]]);
jacobian_matrix;
```

```
[2*u*v - 208/63*v*w + 37/21*w^2   2*u*w - 67/27*v*w + 13/9*w^2   v^2 - 85/21*v*w
+ 13/7*w^2]
[u^2 - 208/63*u*w + 19/7*v*w - 103/126*w^2   -67/27*u*w + 4/3*v*w - 17/27*w^2
2*u*v - 85/21*u*w + 31/7*v*w - 43/42*w^2]
[-208/63*u*v + 19/14*v^2 + 74/21*u*w - 103/63*v*w   u^2 - 67/27*u*v + 2/3*v^2 +
26/9*u*w - 34/27*v*w   -85/21*u*v + 31/14*v^2 + 26/7*u*w - 43/21*v*w]
```

[33]:
```
det:=Determinant(jacobian_matrix);
D:=Curve(P2uvw,det);
D;
```

```
Curve over Rational Field defined by
-3*u^4*v^2 + 67/9*u^3*v^3 - 2*u^2*v^4 + 85/7*u^4*v*w - 223/7*u^3*v^2*w +
853/42*u^2*v^3*w - 7/2*u*v^4*w - 39/7*u^4*w^2 + 143/7*u^3*v*w^2 - 25*u^2*v^2*w^2
+ 53/6*u*v^3*w^2 - 13/14*v^4*w^2 - 3*u^3*w^3 + 139/14*u^2*v*w^3 -
47/14*u*v^2*w^3 - 10/63*v^3*w^3 - 6/7*u^2*w^4 - 1/14*u*v*w^4 + 3/14*v^2*w^4
```

[34]:
```
C:=phi(D);
```

[35]:
```
C;
```

```
Curve over Rational Field defined by
x^4 - 170/21*x^3*y + 8863/441*x^2*y^2 - 2210/147*x*y^3 + 169/49*y^4 -
134/27*x^3*z + 8140/567*x^2*y*z - 22844/1323*x*y^2*z + 2440/441*y^3*z +
5461/729*x^2*z^2 - 23084/1701*x*y*z^2 + 26758/3969*y^2*z^2 - 268/81*x*z^3 +
598/189*y*z^3 + 4/9*z^4
```

[36]:
```
phires:=Restriction(phi,D,C);
```

[37]:
```
phires;
```

```
Mapping from: CrvPln: D to CrvPln: C
with equations :
u^2*v - 208/63*u*v*w + 19/14*v^2*w + 37/21*u*w^2 - 103/126*v*w^2
u^2*w - 67/27*u*v*w + 2/3*v^2*w + 13/9*u*w^2 - 17/27*v*w^2
u*v^2 - 85/21*u*v*w + 31/14*v^2*w + 13/7*u*w^2 - 43/42*v*w^2
```

[38]:
```
IsInvertible(phires);
```

```
true Mapping from: CrvPln: C to CrvPln: D
with equations :
-392/1755*x^4 + 1904/1053*x^3*y - 126013/31590*x^2*y^2 + 1513/1134*x*y^3 +
299/1890*y^4 + 52528/47385*x^3*z - 108178/28431*x^2*y*z + 477419/94770*x*y^2*z -
15394/22113*y^3*z - 390040/255879*x^2*z^2 + 1668814/426465*x*y*z^2 -
408167/142155*y^2*z^2 + 52528/142155*x*z^3 - 29092/47385*y*z^3
-392/1755*x^3*z + 7126/5265*x^2*y*z - 1499/1215*x*y^2*z + 338/405*y^3*z +
52528/47385*x^2*z^2 - 220094/142155*x*y*z^2 - 4943/9477*y^2*z^2 -
```

```
390040/255879*x*z^3 + 600992/426465*y*z^3 + 52528/142155*z^4
y^3*z - 140/117*y^2*z^2 + 1568/5265*y*z^3
and inverse
u^2*v - 208/63*u*v*w + 19/14*v^2*w + 37/21*u*w^2 - 103/126*v*w^2
u^2*w - 67/27*u*v*w + 2/3*v^2*w + 13/9*u*w^2 - 17/27*v*w^2
u*v^2 - 85/21*u*v*w + 31/14*v^2*w + 13/7*u*w^2 - 43/42*v*w^2
```

[39]:
```
Extend(Inverse(phires));
```

```
Mapping from: CrvPln: C to CrvPln: D
with equations :
-392/1755*x^4 + 1904/1053*x^3*y - 126013/31590*x^2*y^2 + 1513/1134*x*y^3 +
299/1890*y^4 + 52528/47385*x^3*z - 108178/28431*x^2*y*z + 477419/94770*x*y^2*z -
15394/22113*y^3*z - 390040/255879*x^2*z^2 + 1668814/426465*x*y*z^2 -
408167/142155*y^2*z^2 + 52528/142155*x*z^3 - 29092/47385*y*z^3
-392/1755*x^3*z + 7126/5265*x^2*y*z - 1499/1215*x*y^2*z + 338/405*y^3*z +
52528/47385*x^2*z^2 - 220094/142155*x*y*z^2 - 4943/9477*y^2*z^2 -
390040/255879*x*z^3 + 600992/426465*y*z^3 + 52528/142155*z^4
y^3*z - 140/117*y^2*z^2 + 1568/5265*y*z^3
and inverse
u^2*v - 208/63*u*v*w + 19/14*v^2*w + 37/21*u*w^2 - 103/126*v*w^2
u^2*w - 67/27*u*v*w + 2/3*v^2*w + 13/9*u*w^2 - 17/27*v*w^2
u*v^2 - 85/21*u*v*w + 31/14*v^2*w + 13/7*u*w^2 - 43/42*v*w^2
and alternative equations :
1/2*x^3 - 73/24*x^2*y + 1979/392*x*y^2 - 741/392*y^3 - 67/54*x^2*z +
2105/504*x*y*z - 494/147*y^2*z + 1/3*x*z^2 - 19/28*y*z^2
1/2*x^2*z - 1/84*x*y*z - 13/14*y^2*z - 67/54*x*z^2 + 299/252*y*z^2 + 1/3*z^3
x*y*z - 57/28*y^2*z

85/42*x^3 - 71075/7056*x^2*y + 187255/16464*x*y^2 - 19175/5488*y^3 +
515/756*x^2*z + 27455/7056*x*y*z - 15265/4116*y^2*z - 1213/756*x*z^2 +
1871/3528*y*z^2 + 13/42*z^3
113/28*x^2*z - 13105/3528*x*y*z - 13/588*y^2*z - 10831/2268*x*z^2 +
37595/10584*y*z^2 + 8/9*z^3
x^2*z - 747/392*y^2*z - 67/27*x*z^2 + 299/126*y*z^2 + 2/3*z^3

67/54*x^4 - 4583/1512*x^3*y + 265/392*x^2*y^2 + 2791/2744*x*y^3 - 209/343*y^4 -
4975/1458*x^3*z + 56167/4536*x^2*y*z - 73127/5292*x*y^2*z + 103759/18522*y^3*z +
134/81*x^2*z^2 - 1871/378*x*y*z^2 + 5681/1764*y^2*z^2 - 2/9*x*z^3 + 19/42*y*z^3
1/2*x^4 - 57/28*x^3*y + 85/1764*x^2*y^2 + 2197/588*x*y^3 - 169/98*y^4 -
67/54*x^3*z + 14081/2268*x^2*y*z - 37609/5292*x*y^2*z + 3887/1764*y^3*z +
1/3*x^2*z^2 - 85/63*x*y*z^2 + 13/21*y^2*z^2
x^3*y - 73/12*x^2*y^2 + 1979/196*x*y^3 - 741/196*y^4
```

[40]:
```
IsNonsingular(C);
```

```
true
```

1.7 Some things Magma can do that are hard to find in other computer algebra packages

- Computations with finite groups, such as their character tables, cohomology, subgroups
- Computations with elliptic curves over number fields

[41]:
```
G:=PSL(2,7);G;
```

```
Permutation group G acting on a set of cardinality 8
Order = 168 = 2^3 * 3 * 7
(3, 6, 7)(4, 5, 8)
(1, 8, 2)(4, 5, 6)
```

[42]:
```
CharacterTable(G);
```

```
Character Table of Group G
--------------------------


------------------------------
Class |   1  2  3  4    5    6
Size  |   1 21 56 42   24   24
Order |   1  2  3  4    7    7
------------------------------
p  =  2   1  1  3  2    5    6
p  =  3   1  2  1  4    6    5
p  =  7   1  2  3  4    1    1
------------------------------
X.1   +   1  1  1  1    1    1
X.2   0   3 -1  0  1   Z1 Z1#3
X.3   0   3 -1  0  1 Z1#3   Z1
X.4   +   6  2  0  0   -1   -1
X.5   +   7 -1  1 -1    0    0
X.6   +   8  0 -1  0    1    1


Explanation of Character Value Symbols
--------------------------------------

# denotes algebraic conjugation, that is,
#k indicates replacing the root of unity w by w^k

Z1     = (CyclotomicField(7: Sparse := true)) ! [ RationalField() | 0, 1, 1, 0,
1, 0 ]
```

[43]:
```
SubgroupLattice(G);
```

```
Partially ordered set of subgroup classes
-----------------------------------------

[15]  Order 168  Length 1   Maximal Subgroups: 9 13 14
---
[14]  Order 24   Length 7   Maximal Subgroups: 8 10 11
[13]  Order 24   Length 7   Maximal Subgroups: 8 10 12
---
[12]  Order 12   Length 7   Maximal Subgroups: 3 5
[11]  Order 12   Length 7   Maximal Subgroups: 3 6
[10]  Order 8    Length 21  Maximal Subgroups: 5 6 7
---
[ 9]  Order 21   Length 8   Maximal Subgroups: 3 4
[ 8]  Order 6    Length 28  Maximal Subgroups: 2 3
[ 7]  Order 4    Length 21  Maximal Subgroups: 2
[ 6]  Order 4    Length 7   Maximal Subgroups: 2
[ 5]  Order 4    Length 7   Maximal Subgroups: 2
---
[ 4]  Order 7    Length 8   Maximal Subgroups: 1
[ 3]  Order 3    Length 28  Maximal Subgroups: 1
[ 2]  Order 2    Length 21  Maximal Subgroups: 1
---
[ 1]  Order 1    Length 1   Maximal Subgroups:
```

[44]:
```
M:=TrivialModule(G,GF(2));
ch:=CohomologyModule(G,M);
```

[45]:
```
CohomologyGroup(ch,2);
```

```
Full Vector space of degree 1 over GF(2)
```

[46]:
```
R<x>:=PolynomialRing(Rationals());
C:=GenusOneModel(2*(x^4-17));
```

[47]:
```
IsLocallySolvable(C,17);
```

```
true (1 + O(17^20) : O(17^20) : -6211390105882224290024 58 + O(17^20))
```

[48]:
```
E:=Jacobian(C);E;
```

```
Elliptic Curve defined by y^2 = x^3 + 272*x over Rational Field
```

[49]:
```
Rank(E);
```

```
0 true
```

[50]:
```
TwoSelmerGroup(E);
```

```
Abelian Group isomorphic to Z/2 + Z/2 + Z/2
Defined on 3 generators
Relations:
2*$.1 = 0
2*$.2 = 0
2*$.3 = 0
Mapping from: Univariate Quotient Polynomial Algebra in theta over Rational
Field
with modulus theta^3 + 17*theta to Abelian Group isomorphic to Z/2 + Z/2 + Z/2
Defined on 3 generators
Relations:
2*$.1 = 0
2*$.2 = 0
2*$.3 = 0 given by a rule
```

[51]:
```
Rank(QuadraticTwist(E,2));
```

```
2 true
```

[52]:
```
Rank(BaseChange(E,QuadraticField(2)));
```

```
2 true
```

[ ]:
```

```