In [45]:
```
p = random_prime(2^16, 2^14)
while p % 4 != 1:
    p = random_prime(2^16, 2^14)
print(p)
```

```
62617
```

In [46]:
```
Zp = IntegerModRing(p)
```

In [51]:
```
a = Zp.random_element()
print(a)
```

```
30204
```

In [52]:
```
legendre_symbol(a, p)
```

Out[52]:
```
1
```

In [53]:
```
a = 30204
```

In [54]:
```
PolZp = PolynomialRing(Zp, 'x')
PolZp
```

Out[54]:
```
Univariate Polynomial Ring in x over Ring of integers modulo 62617
```

In [55]:
```
pol = PolZp(x^2-a)
pol
```

Out[55]:
```
x^2 + 32413
```

In [56]:
```
R = PolynomialQuotientRing(PolZp, pol, 'xx')
R
```

Out[56]:
```
Univariate Quotient Polynomial Ring in xx over Ring of integers modulo 62617 with
modulus x^2 + 32413
```

In [57]:
```
xx = R(x)
```

In [87]:
```
z = Zp.random_element()
while z == 0:
    z = Zp.random_element()
print(z)
```

```
4944
```

In [88]:
```
(1+z*xx)^((p-1)//2)
```

Out[88]:
```
57103*xx
```

In [89]:
```
elem = (1+z*xx)^((p-1)//2)
```

In [90]:
```
v, u = elem[0], elem[1]
v, u
```

Out[90]:
```
(0, 57103)
```

In [91]:
```
sol1 = -v/u
sol2 = (1-v)/u
```

```
sol3 = (-1-v)/u
sol1, sol2, sol3
```

Out[91]:
```
(0, 16364, 46253)
```

In [76]:
```
sol1^2 == a, sol2^2 == a, sol3^2 == a
```

Out[76]:
```
(False, True, True)
```

In [ ]:
```

```