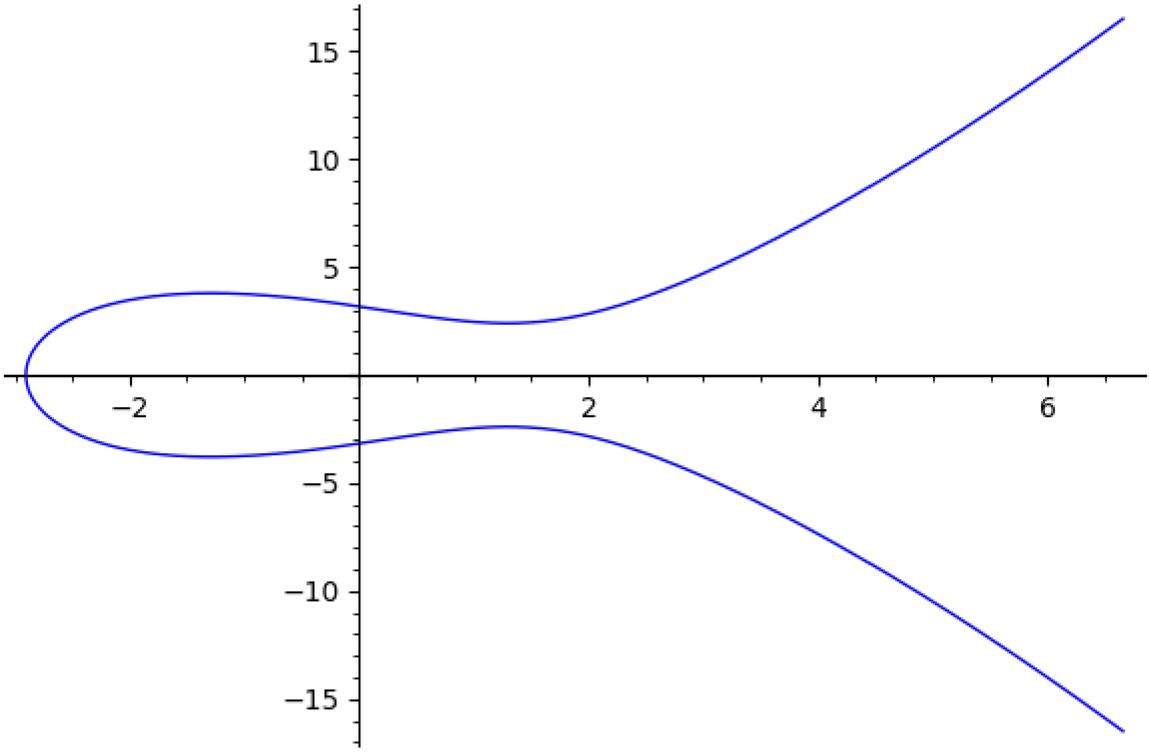
In [9]:
```
E1 = EllipticCurve(QQ, [-5, 10])
E1
```

Out[9]:
```
Elliptic Curve defined by y^2 = x^3 - 5*x + 10 over Rational Field
```

In [10]:
```
E1.plot()
```

Out[10]:

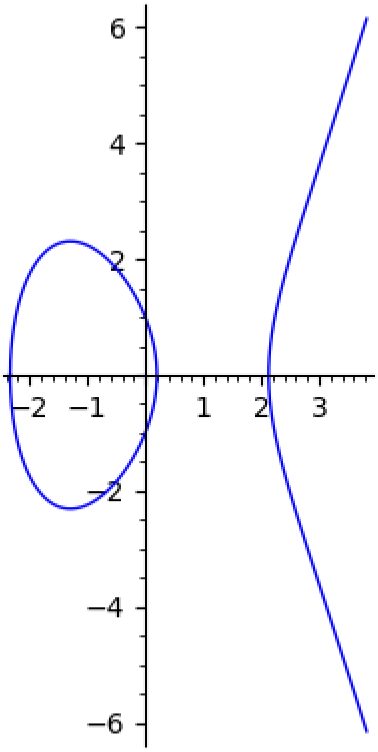


In [11]:
```
E2 = EllipticCurve(QQ, [-5, 1])
E2
```

Out[11]:
```
Elliptic Curve defined by y^2 = x^3 - 5*x + 1 over Rational Field
```

In [15]:
```
E2.plot(aspect_ratio=true)
```

Out[15]:



In [49]:
```
P=E2.an_element()
P
```

Out[49]: (0 : 1 : 1)

In [50]:
```
P+P
```

Out[50]: (25/4 : 117/8 : 1)

In [16]:
```
Zp = IntegerModRing(23)
Zp
```

Out[16]: Ring of integers modulo 23

In [17]:
```
a = Zp(2)
b = Zp(5)
```

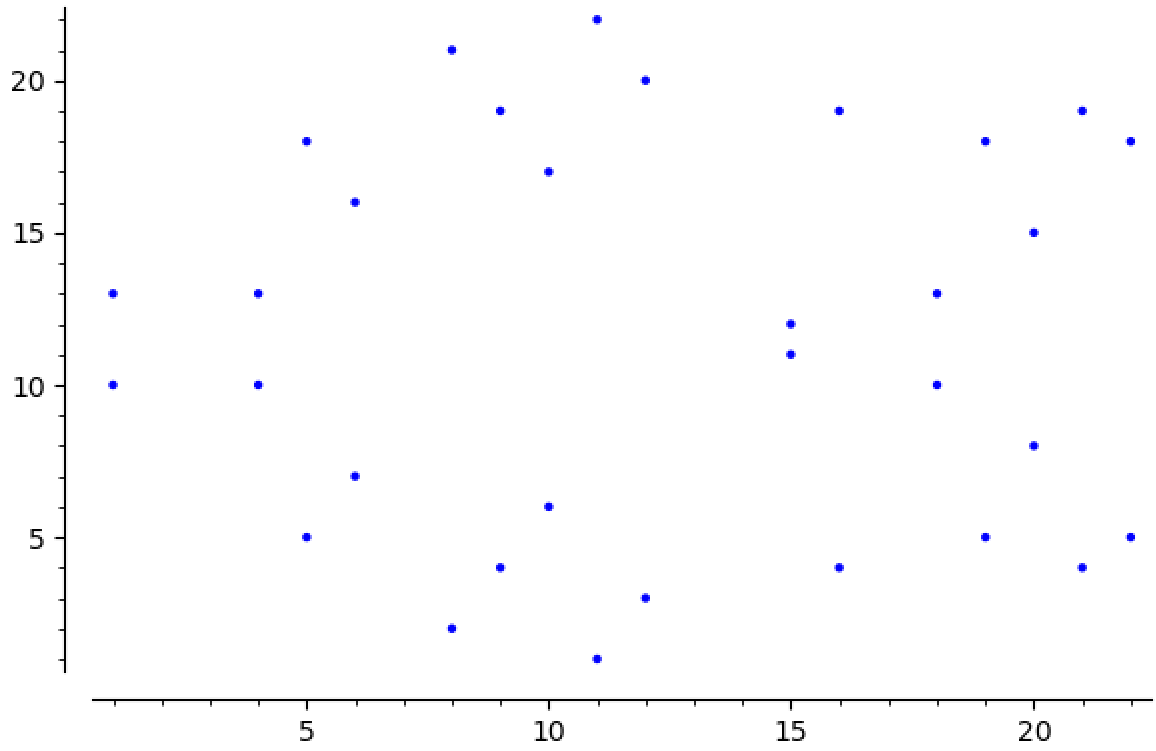
In [18]:
```
E = EllipticCurve(Zp, [a, b])
```

In [19]:
```
E
```

Out[19]: Elliptic Curve defined by y^2 = x^3 + 2*x + 5 over Ring of integers modulo 23

In [20]:
```
E.plot()
```

Out[20]:



In [21]:
```
E
```

Out[21]: Elliptic Curve defined by y^2 = x^3 + 2*x + 5 over Ring of integers modulo 23

In [23]:
```
legendre_symbol(8, 23)
```

Out[23]: 1

In [24]:
```
sqrt(Zp(8))
```

Out[24]: 10

In [25]:
```
legendre_symbol(17, 23)
```

Out[25]: -1

In [27]:
```
P = E.an_element()
P
```

Out[27]: (21 : 4 : 1)

In [28]:
```
P+P
```

Out[28]: (20 : 15 : 1)

In [29]:
```
2*P
```

Out[29]: (20 : 15 : 1)

In [30]:
```
3*P
```

Out[30]: (11 : 1 : 1)

In [32]:
```
E.order()
```

Out[32]: 33

In [37]:
```
E(0,1,0) + P
```

Out[37]:
```
(21 : 4 : 1)
```

In [38]:
```
P.order()
```

Out[38]: 33

In [39]:
```
33*P
```

Out[39]:
```
(0 : 1 : 0)
```

In [40]:
```
3*P
```

Out[40]:
```
(11 : 1 : 1)
```

In [41]:
```
11*P
```

Out[41]:
```
(4 : 13 : 1)
```

In [42]:
```
[k*P for k in range(1, 34)]
```

Out[42]:
```
[(21 : 4 : 1),
 (20 : 15 : 1),
 (11 : 1 : 1),
 (18 : 13 : 1),
 (16 : 4 : 1),
 (9 : 19 : 1),
 (19 : 5 : 1),
 (12 : 3 : 1),
 (15 : 12 : 1),
 (22 : 5 : 1),
 (4 : 13 : 1),
 (6 : 7 : 1),
 (8 : 21 : 1),
 (10 : 17 : 1),
 (1 : 10 : 1),
 (5 : 5 : 1),
 (5 : 18 : 1),
 (1 : 13 : 1),
 (10 : 6 : 1),
 (8 : 2 : 1),
 (6 : 16 : 1),
 (4 : 10 : 1),
 (22 : 18 : 1),
 (15 : 11 : 1),
 (12 : 20 : 1),
 (19 : 18 : 1),
 (9 : 4 : 1),
 (16 : 19 : 1),
 (18 : 10 : 1),
 (11 : 22 : 1),
 (20 : 8 : 1),
 (21 : 19 : 1),
 (0 : 1 : 0)]
```

In [43]:
```
Q = E(6, 16)
```

In [44]:
```
# diffie-hellmann
```

In [45]:
```
a = 12
P_Alice = a*P
b = 16
P_Bob = b*P
```

In [46]:
```
a*(P_Bob), b*(P_Alice)
```

Out[46]:
```
((9 : 4 : 1), (9 : 4 : 1))
```

In [51]:
```
n = 35
Zn = IntegerModRing(n)
```

In [52]:
```
x0 = Zn.random_element()
y0 = Zn.random_element()
```

In [53]:
```
a = Zn.random_element()
b = y0^2-x0^3-a*x0
```

In [54]:
```
gcd(n, 4*a^3+27*b^2)
```

Out[54]:
```
1
```

In [55]:
```
E = EllipticCurve(Zn, [a, b])
E
```

Out[55]:
```
Elliptic Curve defined by y^2 = x^3 + 26*x + 29 over Ring of integers modulo 35
```

In [56]:
```
P = E(x0, y0)
P
```

Out[56]:
```
(5 : 33 : 1)
```

In [57]:
```
P+P
```

Out[57]:
```
(26 : 16 : 1)
```

In [58]:
```
3*P
```