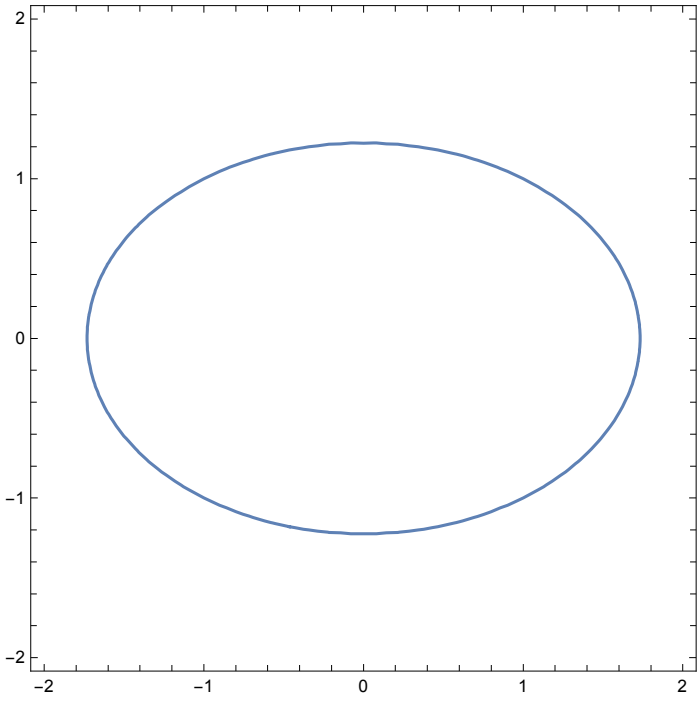
In[1]:=
```
(*Лабораторная работа №2*)
(*По курсу «Защита информационных процессов в компьютерных системах»*)
(*Исследование свойств эллиптических кривых.*)

(*
    Кутузов Илья
      А-12м-20
*)

(*Задание 1*)
(*Построить график эллипса X^2+2*Y^2=3, используя ContourPlot[] пакета математки*)
```
контурный график
```
AbsScaleX1 = 2;
AbsScaleY1 = 2;
ContourPlot[x^2 + 2 * y^2 == 3,
```
контурный график
```
 {x, -AbsScaleX1, AbsScaleX1}, {y, -AbsScaleY1, AbsScaleY1}]
```

Out[3]=

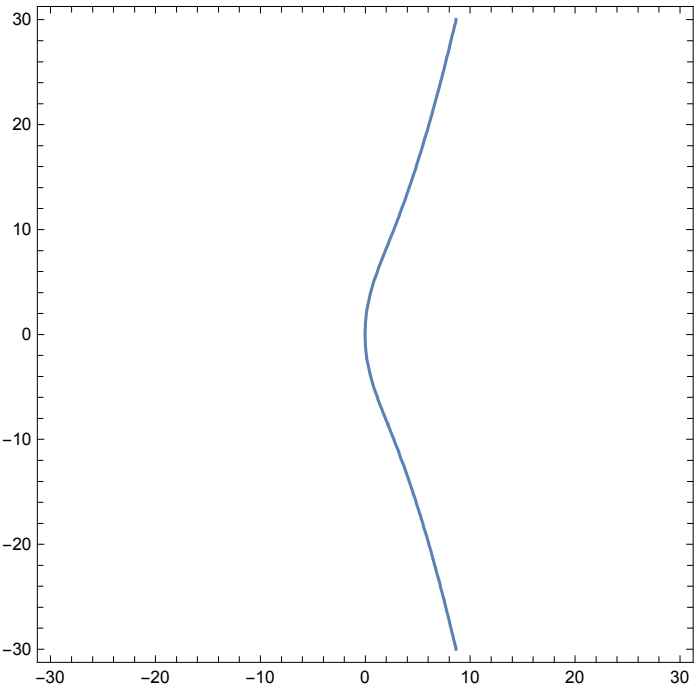


In[4]:=
```
(*Задание 2*)
(*В поле рациональных чисел построить графики эллиптических кривых Y^2=
 X^3+a*X+b для положительного и отрицательного коэффициента a*)
a = 29;
b = 1;
c = 0
p = 31;
n = 7;
AbsScaleX2 = 30;
AbsScaleY2 = 30;
ContourPlot[y^2 == x^3 + a * x + b,
 {x, -AbsScaleX2, AbsScaleX2}, {y, -AbsScaleY2, AbsScaleY2}]
ContourPlot[y^2 == x^3 - a * x + b, {x, -AbsScaleX2, AbsScaleX2},
 {y, -AbsScaleY2, AbsScaleY2}]
```
контурный график

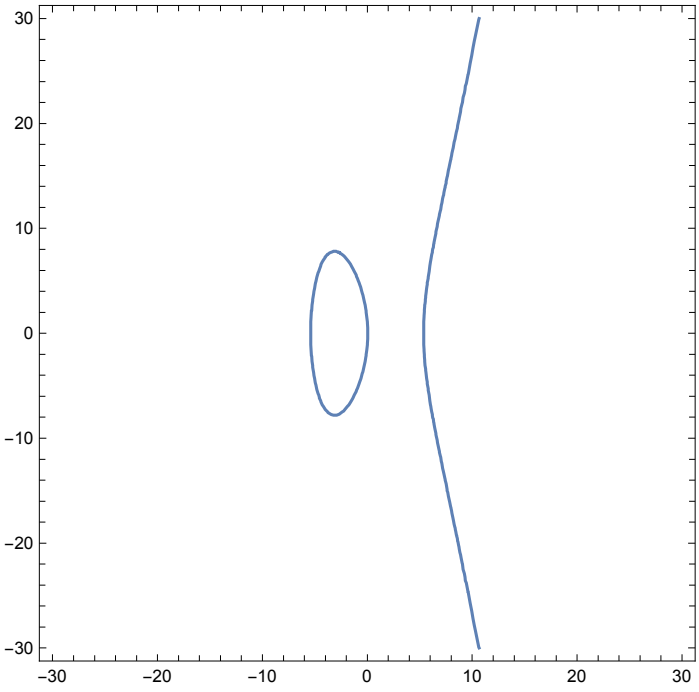
контурный график

Out[6]= 0

Out[11]=



Out[12]=



In[13]:=
```
(*Проверить выполенение условия гладкости кривой -16(4*a^3+27*b^2)≠0*)
-16 * (4 * a^3 + 27 * b^2) ≠ 0
```

Out[13]= True

In[14]:=
```
(*Проверить, является ли заданный в правой части уравнения многочлен неприводимым,
используя функцию Factor[]*)
```
факторизовать
```
Factor[x^3 + a * x + b]
```
факторизовать

Out[14]= $1 + 29\,x + x^3$

In[15]:=
```
(*Задание 3*)
(*Проверить выполнение условия гладкости кривой в GF(p)*)
```

In[16]:=
```
Mod[4 * a^3 + 27 * b^2, 83] ≠ 0
```
остаток от деления

Out[16]= True

In[17]:=
```
(*Определить число точек заданной кривой в поле GF(p)... *)
Clear[x, y];
```
очистить
```
g1 = {x, y} /. Flatten[Table[
```
уплостить | таблица значений
```
    FindInstance[y^2 == x^3 + a * x + b  && x == u, {x, y}, 2, Modulus → p], {u, 0, p - 1}], 1]
```
найти частный случай | модуль

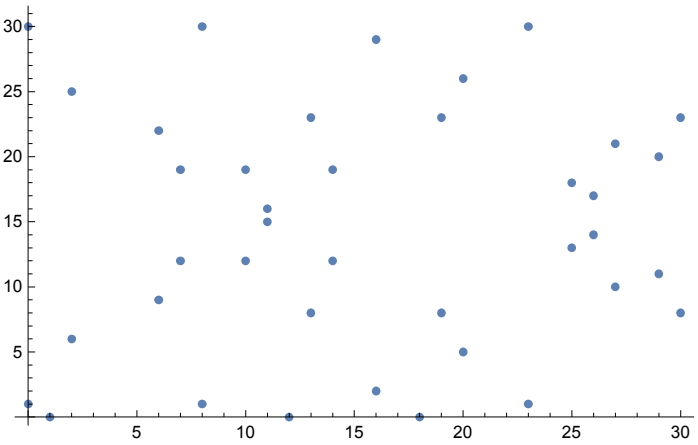
Out[18]=
```
{{0, 1}, {0, 30}, {1, 0}, {2, 6}, {2, 25}, {6, 9}, {6, 22}, {7, 12},
 {7, 19}, {8, 1}, {8, 30}, {10, 12}, {10, 19}, {11, 15}, {11, 16}, {12, 0},
 {13, 8}, {13, 23}, {14, 12}, {14, 19}, {16, 2}, {16, 29}, {18, 0}, {19, 8},
 {19, 23}, {20, 5}, {20, 26}, {23, 1}, {23, 30}, {25, 13}, {25, 18}, {26, 14},
 {26, 17}, {27, 10}, {27, 21}, {29, 11}, {29, 20}, {30, 8}, {30, 23}}
```

In[19]:=
```
(*Длина полученного списка с учетом точки в бесконечности-порядок кривой*)
Length[g1] + 1
```
длина

Out[19]= 40

In[20]:=
```
(*... и построить точеченый график*)
ListPlot[g1]
```
диаграмма разброса данных

Out[20]=



In[21]:=
```
(*Задание 4*)
(*Сложить две точки, принадлежащие заданной эллиптической кривой,
зафиксировать полученный результат на точеченом графике*)
(*Операцию сложения можно выполнить, используя следующий программный модуль*)
(*При использовании данного модуля следует учитывать,
что он реализует сложение точек эллиптических кривых вида: y^2=x^3+ax^2+bx+c*)
```

In[22]:=
```
EllipticAdd[p_, a_, b_, c_, P_List, Q_List] := Module[{lam, x3, y3, P3},
                                                 программный модуль
  Which[
  условный оператор с множественными ветвями
   P == {O}, Q,
        O большое
   Q == {O}, P,
        O большое
   P[[1]] ≠ Q[[1]],
       lam = Mod[(Q[[2]] - P[[2]]) PowerMod[Q[[1]] - P[[1]], p - 2, p], p];
             остаток от деления      степень по модулю
       x3 = Mod[lam^2 - a - P[[1]] - Q[[1]], p];
            остаток от деления
       y3 = Mod[-(lam (x3 - P[[1]]) + P[[2]]), p];
            остаток от деления
       {x3, y3},
   (P == Q) ∧ (P[[2]] == 0), {O},
                             O большое
   (P == Q) ∧ (P ≠ {O}),
              O большое
       lam = Mod[(3 * P[[1]]^2 + 2 a * P[[1]] + b) PowerMod[2 P[[2]], p - 2, p], p];
             остаток от деления                    степень по модулю
       x3 = Mod[lam^2 - a - P[[1]] - Q[[1]], p];
            остаток от деления
       y3 = Mod[-(lam (x3 - P[[1]]) + P[[2]]), p];
            остаток от деления
       {x3, y3},
   (P[[1]] == Q[[1]]) ∧ (P[[2]] ≠ Q[[2]]), {O}
                                          O большое
  ]
 ]
EllipticAdd[a, b, c, p, g1[[4]], g1[[2]]]
```

Out[23]= {25, 9}

In[24]:=
```
(*Задание 5*)
(*Провести тестирование операции сложения,повторив следующие действия*)

ptest = 11;
atest = 0;
btest = 6;
ctest = 3;

{EllipticAdd[ptest, atest, btest, ctest, {4, 6}, {9, 4}],
 EllipticAdd[ptest, atest, btest, ctest, {9, 4}, {9, 4}],
 EllipticAdd[ptest, atest, btest, ctest, {4, 6}, {4, 6}],
 EllipticAdd[ptest, atest, btest, ctest, {4, 6}, {O}],
                                                 O большое
 EllipticAdd[ptest, atest, btest, ctest, {4, 6}, {4, 5}],
 EllipticAdd[ptest, atest, btest, ctest, {O}, {9, 4}]}
                                         O большое
```

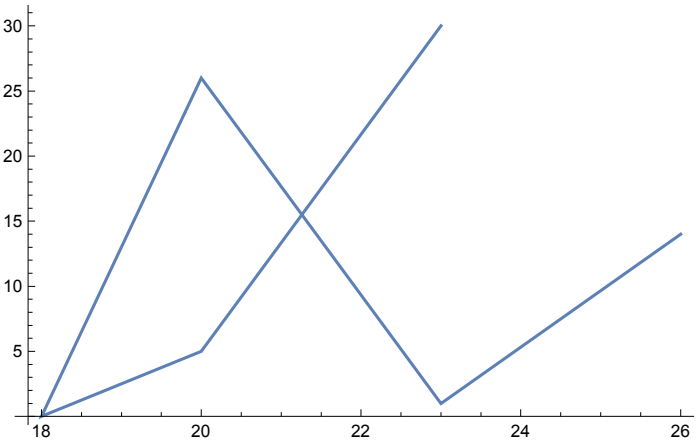
Out[28]= {{3, 9}, {7, 6}, {4, 5}, {4, 6}, {O}, {9, 4}}

In[29]:=
```
(*Задание 7*)
(*Провести операцию умножения произвольной
 точки на число n (Табл.1) и построить граф переходов.*)

Mult[p1_, a1_, b1_, c1_, point1_, num1_] :=
  Module[{p = p1, a = a1, b = b1, c = c1, point = point1, num = num1}, temp = point;
  программный модуль
   q = {};
   Do[temp = EllipticAdd[p, a, b, c, point, temp];
   оператор цикла
    AppendTo[q, temp], {i, 2, num}];
    добавить в конец к
   q];
path = Mult[p, 0, a, b, g1[[8]], n]
ListLinePlot[path]
линейный график данных
```

Out[30]=
```
{{26, 14}, {23, 1}, {20, 26}, {18, 0}, {20, 5}, {23, 30}}
```

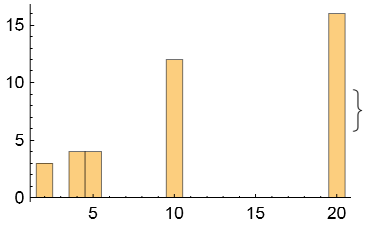
Out[31]=



In[32]:=
```
(*Задание 8*)
(*Для каждой точки заданной кривой
 определить её порядок (Определение:Порядком точки P
      эллиптической кривой называется наименьшее натуральное число m≠0,
   для которого mP=O.См.также articles\osnovy_elliptic.pdf,page 69).Построить
 гистограмму распределения порядков точек.*)
Mult2[p1_, a1_, b1_, c1_, point1_, num1_] :=
  Module[{p = p1, a = a1, b = b1, c = c1, point = point1, num = num1}, temp = point;
  программный модуль
   q = {};
   Do[temp = EllipticAdd[p, a, b, c, point, temp];
   оператор цикла
    AppendTo[q, temp], {i, 2, num}];
    добавить в конец к
   temp];
```

In[33]:=
```
s = {}; i = 1;
For[j = 1, j ≤ Length[g1], j++, {
 цикл ДЛЯ   длина
  While[Mult2[p, 0, a, b, g1[[j]], i] !=  {O}, i++], AppendTo[s, i], i = 1}];
   цикл–пока                                 О большое  добавить в конец к
{Tally[s], Histogram[s, Length[g1]]}
 подсчитать  гистограмма   длина
```

Out[34]=
```
{{{4, 4}, {2, 3}, {10, 12}, {20, 16}, {5, 4}},
```




```
}
```