Talk: The Birch and Swinnerton-Dyer Conjecture

The Birch and Swinnerton-Dyer Conjecture:

An Unsolved Problems with Roots in Ancient Times

Birch and Swinnerton-Dyer in 2000 in Holland

Nonsingular Plane Curves

A nonsingular plane algebraic curve is the set of solutions to a (nonsingular) polynomial:

$$F(X, Y) = 0$$

A *rational point* is $(x, y) \in \mathbf{Q} \times \mathbf{Q}$ such that $F(x, y) = 0$.

- **Theorem (old):** A curve of **degree** ≤ 2 has no rational points ($x^2 + y^2 = -1$) or infinitely many rational points ($x^2 + y^2 = 1$), and there is a way to decide which and enumerate all solutions.
- **Faltings Theorem (1985):** A curve of **degree** ≥ 4 has finitely many rational points.
- **Birch and Swinnerton-Dyer Conjecture (1960s):** A curve of **degree 3** has either finitely many rational

points ($x^3 + y^3 = 1$) or infinitely many rational points $y^2 + y = x^3 - x$. The BSD Conjecture provides a way to decide which and enumerate all solutions.

Rational Points on Plane Curves

We find rational points on the curve you type in.

```
@interact
def f(F = ('0 = F(x,y) = ', 'x^2 + y^2 - 1'),
          search_bound=selector([1..10], buttons=True), box=(1..20), square=
('Square aspect ratio', False)) :
   R.<x,y> = QQ[]
   try: F = R(F.lower())
   except: print "Enter a polynomial in x, y with rational coefficients.";
return
   C = Curve(F)
   P = C.rational_points(bound=search_bound)
   show(tuple(P))
   eps = 0.1
   xmax = max([box]+[p[0] for p in P])+eps; ymax = max([box]+[p[1] for p in
P])+eps
   xmin = min([-box]+[p[0] for p in P])-eps; ymin = min([-box]+[p[1] for p in
P])-eps
   g = implicit_plot(F, (x,xmin,xmax), (y,ymin,ymax), plot_points=300)
   if len(P) > 0: g += points(P,pointsize=40)
   if square: show(g, aspect_ratio=1, figsize=5)
   else: show(g)
```

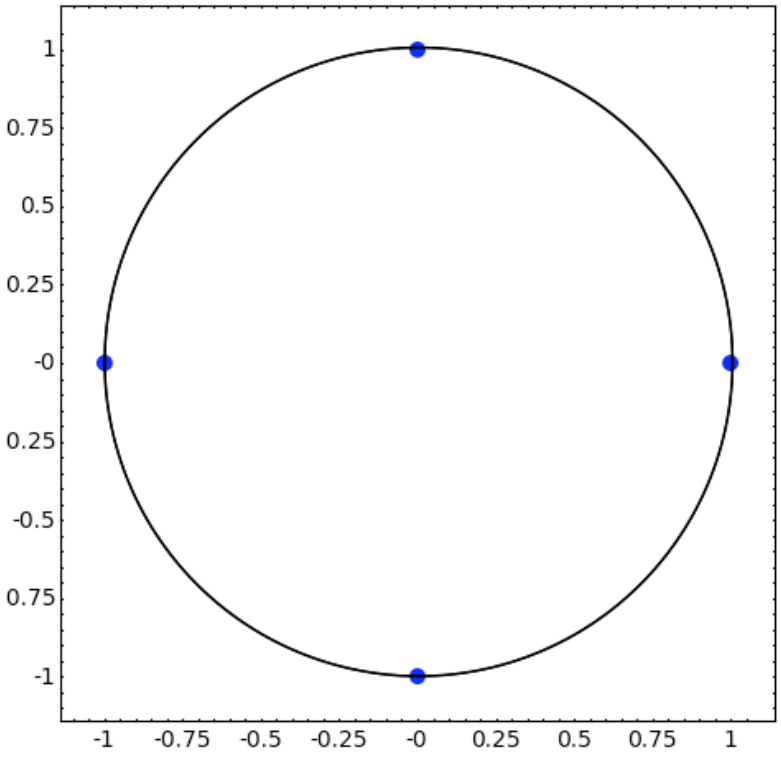
0 = F(x,y) = x^2 + y^2 - 1

search_bound 1 2 3 4 5 6 7 8 9 10

box

Square aspect ratio

$$((1, 0), (-1, 0), (0, 1), (0, -1))$$


1
0.75
0.5
0.25
-0
0.25
-0.5
0.75
-1
-1
-0.75
-0.5
-0.25
-0
0.25
0.5
0.75
1

Enumerating Pythagorean Triples

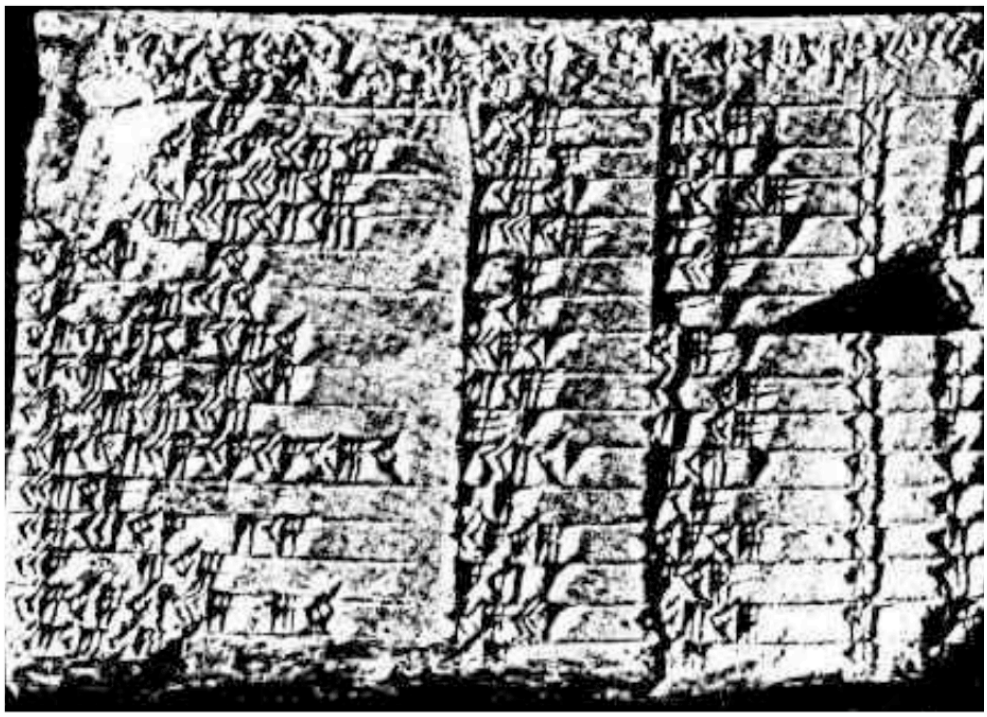
Ancient Problem: Find all solutions $x, y \in \mathbf{Q}$ to the equation $x^2 + y^2 = 1$. Equivalently, by clearing denominators, find all Pythagorean triples (a, b, c) such that $a^2 + b^2 = c^2$. This problem goes back **thousands of years**!

Babylonians

1800-1600 B.C.

$$(3,4,5)$$
$$(5,12,13)$$
$$(7,24,25)$$
$$(9,40,41)$$
$$(11,60,61)$$
$$(13,84,85)$$
$$(15,8,17)$$
$$(21,20,29)$$
$$(33,56,65)$$
$$(35,12,37)$$
$$(39,80,89)$$
$$(45,28,53)$$
$$(55,48,73)$$
$$(63,16,65)$$
$$(65,72,97)$$
$$(77,36,85)$$
$$\vdots$$

Pythagorean Triples



Triples of whole numbers *a*, *b*, *c* such tha

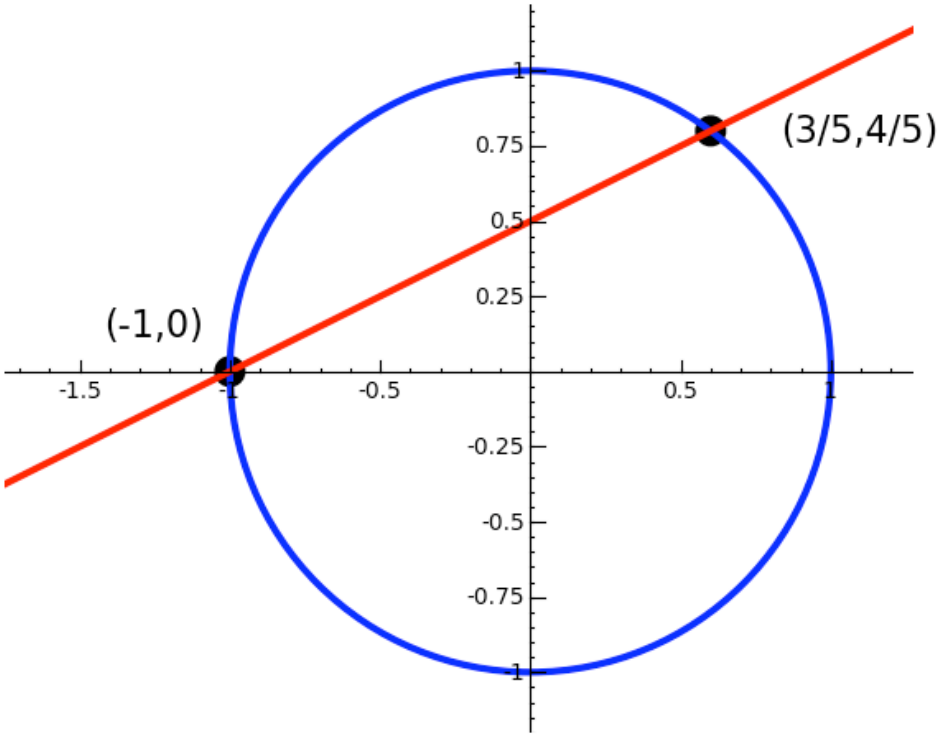
$$a^2 + b^2 = c^2$$

Enumerating Pythagorean Triples

There is a nice construction that completely solves this problem. Just draw a line of rational slope through (-1,0) and find the unique other point of intersection with the circle. It will have to be rational, as you can verify with some algebra. Moreover, this gives every point! If (x, y) is any rational solution, then the line through $(-1, 0)$ and (x, y) has rational slope $\frac{y}{x+1}$, so we would find it via the above process.

```
G = circle((0,0),1, rgbcolor='blue', thickness=3)
G += point([(-1,0), (3/5,4/5)], pointsize=150, rgbcolor='black')
G += line([(-1-1,0-1/2), (3/5+1,4/5+1/2)], rgbcolor='red', thickness=3)
```

```
G += text("(3/5,4/5)", (3/5+.5,4/5), rgbcolor='black',fontsize=16)
G += text("(-1,0)", (-1.25,0.15), rgbcolor='black',fontsize=16)
G.show(aspect_ratio=1,xmin=-1.5,xmax=1, ymin=-1,ymax=1)
```



Enumerating Pythagorean Triples: Live Demo

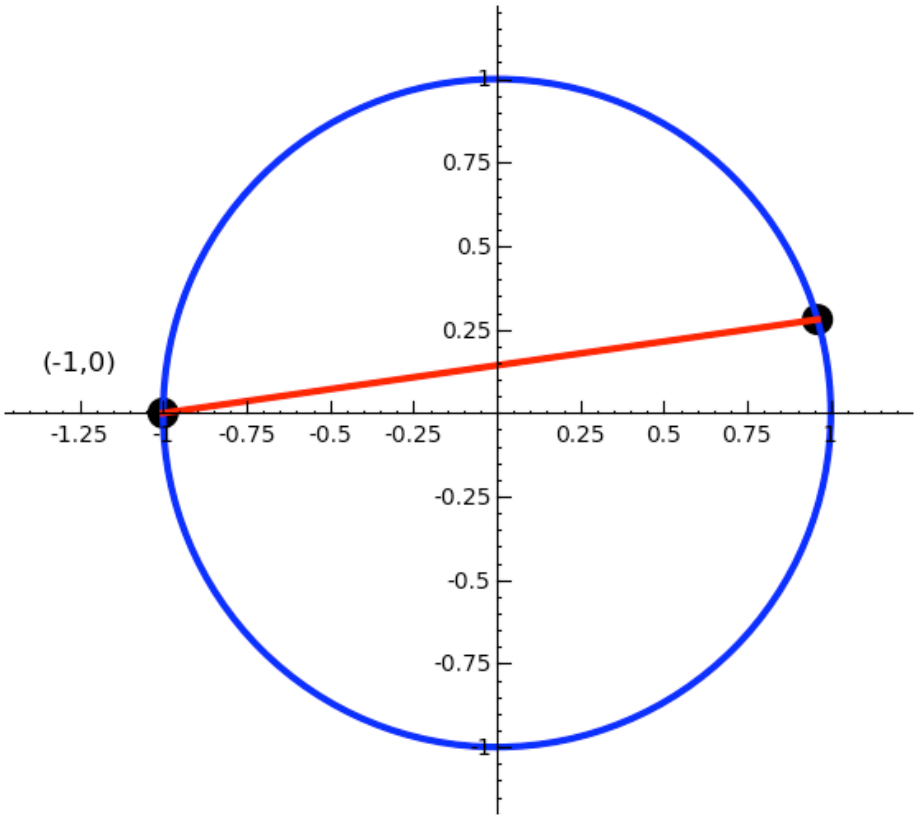
```
@interact
def __(t=('slope',1/2)):
   t = QQ(t)
   x = (1-t^2)/(1+t^2)
   y = 2*t/(1+t^2)
   r = t.numerator()
   s = t.denominator()
   a = s^2 - r^2; b = 2*r*s; c = s^2 + r^2
   html('<h1 align=center>Point (x,y) = $%s$'%latex((x,y)))
   html('Pythagorean (a,b,c) = $%s$</h1>'%latex((a,b,c)))
   G = circle((0,0),1, rgbcolor='blue', thickness=3)
   G += point([(-1,0), (x,y)], pointsize=150, rgbcolor='black')
```

```
G += line([(-1,0), (x,y)], rgbcolor='red', thickness=3)
G += text("(-1,0)", (-1.25,0.15), rgbcolor='black',fontsize=12)
try:
   G.save('a.png',aspect_ratio=1)
except RuntimeError, msg:
    print msg
html('<img src="cell://a.png">')
```

slope `1/2`

Point (x,y) = $\left(\frac{24}{25}, \frac{7}{25}\right)$

Pythagorean (a,b,c) = $(48, 14, 50)$



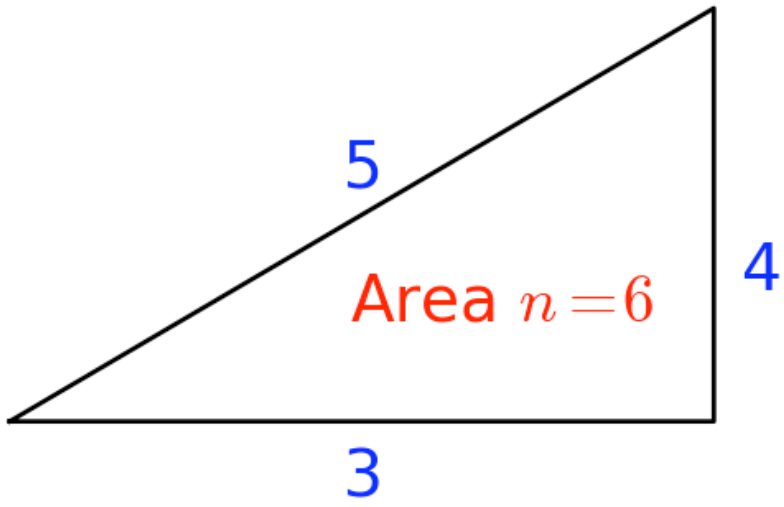
The Congruent Number Problem

Definition: An integer n is a **congruent number** if n is the area of a right triangle with rational side lengths.

Major Unsolved Problem in Mathematics: Give an ***algorithm*** to decide whether or not an integer n is a congruent number.

This is a ***1000-year old open problem***; it is considered by some to be the ***oldest*** open problem in mathematics.

```
T = line([(0,0), (3,0), (3,4), (0,0)],rgbcolor='black',thickness=2)
lbl = text("3",(1.5,-.5),fontsize=28) + text("4",(3.2,1.5),fontsize=28)
lbl += text("5",(1.5,2.5),fontsize=28)
lbl += text("Area $n = 6$", (2.1,1.2), fontsize=28, rgbcolor='red')
show(T+lbl, axes=False)
```



Congruent Numbers and the BSD Conjecture

Theorem: *A proof of the Birch and Swinnerton-Dyer Conjecture would also solve the congruent number problem.*

Proof: Suppose n is a positive integer. Consider the cubic curve $y^2 = x^3 - n^2 x$. Using algebra (see next slide), one sees that this cubic curve has infinitely many rational points if and only if there are rationals a, b, c such that $n = ab/2$ and $a^2 + b^2 = c^2$. The Birch and Swinnerton-Dyer conjecture gives an *algorithm to decide whether or not any cubic curve has infinitely many solutions*.

Explicit Bijection

In fact, there is a bijection between

$$A = \left\{(a,b,c) \in \mathbf{Q}^3 \ : \ \frac{ab}{2} = n,\ a^2 + b^2 = c^2\right\}$$

and

$$B = \left\{(x,y) \in \mathbf{Q}^2 \ : \ y^2 = x^3 - n^2 x, \text{ with } y \neq 0\right\}$$

given by the maps

$$f(a,b,c) = \left(-\frac{nb}{a+c},\ \frac{2n^2}{a+c}\right)$$

and

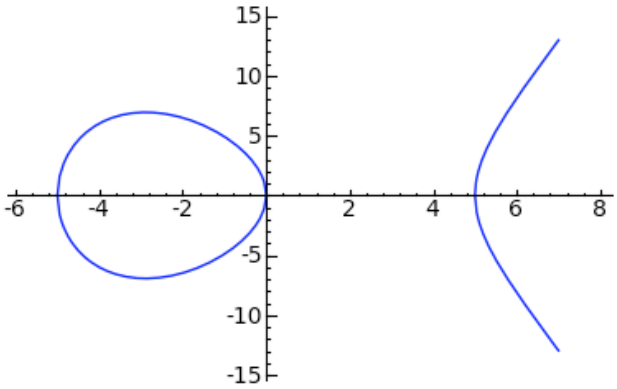
$$g(x,y) = \left(\frac{n^2 - x^2}{y}, -\frac{2xn}{y},\ \frac{n^2 + x^2}{y}\right).$$

5 is a Congruent Number

```
n = 5; x,y = var('x,y')
C = EllipticCurve(y^2 == x^3 - n^2 * x); C
```

```
    Elliptic Curve defined by y^2 = x^3 - 25*x over Rational Field
```

```
show(C.plot(), figsize=4)
```




```
P = C.gens()[0]
print P
print "order of P = ", P.order()
```

```
(-4 : 6 : 1)
order of P =  +Infinity
```

```
(-62279/1728)^2 == (1681/144)^3 - 25*(1681/144)
```

```
True
```

```
x=-4; y=6
```

```

```

```

```

1 is *Not* a Congruent Number

```
n = 1
x,y = var('x,y')
C = EllipticCurve(y^2 == x^3 - n^2 * x)
C
```

```
C.gens()
```

```

```

Which positive integers $n \leq 10$ are congruent numbers?

```
for n in [1..10]:
   print n, EllipticCurve([-n^2,0]).rank() > 0
```

```
1 False
2 False
3 False
4 False
5 True
6 True
```

```
7 True
8 False
9 False
10 False
```

Finding Explicit Rational Right Triangles

```
@interact
def _(n=6, triangles=(1..10), maxtime=(3..30)):
   x,y = var('x,y')
   C = EllipticCurve(y^2 == x^3 - n^2*x)
   try:
       alarm(maxtime)
       t = walltime()
       G = C.gens()
       print "time = %.2f seconds"%walltime(t)
   except RuntimeError:
       print "Sage is unable to provably find generators"
       return
   except KeyboardInterrupt, msg:
       print "Too hard -- timed out after %s seconds"%maxtime
       return
   html("rank = %s\n\n"%len(G))
   if len(G) == 0: print "%s is not a congruent number"%n; return
   def g(x,y,n): return ((n^2-x^2)/y, -2*x*n/y, (n^2+x^2)/y)
   P = G[0]
   html('<h3><font color="red">Rational Right Triangles with Area %s</font>
</h3>'%n)
   for i in [1..triangles]:
       a,b,c = g((i*P)[0], (i*P)[1], n)
```

```
    html("(a,b,c) = $%s$\n"%latex((a,b,c)))
```

n 6

triangles

maxtime

```
time = 2.47 seconds
rank = 2
```

Rational Right Triangles with Area 2009

$(a,b,c) = \left(\frac{280}{3}, \frac{861}{20}, \frac{6167}{60}\right)$

$(a,b,c) = \left(-\frac{3526873}{105720}, -\frac{8669040}{71977}, \frac{950998057921}{7609408440}\right)$

$(a,b,c) = \left(-\frac{47285754253640}{5199355266237}, -\frac{1492214961410019}{3377553875260}, -\frac{776019936413791542813658024 7}{1756110252833226838759662 0}\right)$

The L-function

Let C be a cubic curve (+ a technical condition I'm not mentioning). For each *prime number* p, let N_p be the number of solutions to the cubic modulo p.

Definition: For any cubic curve C, let $a_p = p - N_p$.

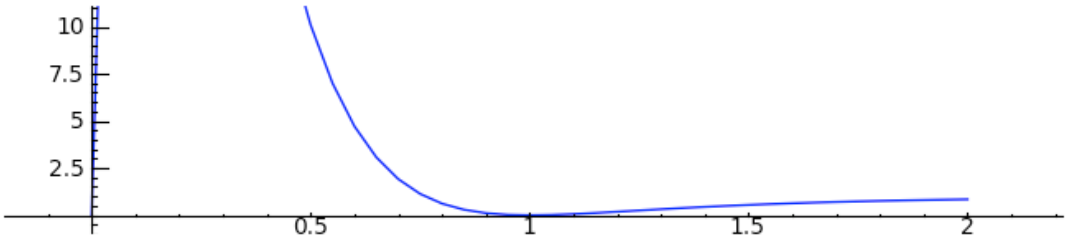
Theorem (Hasse): $|a_p| < 2\sqrt{p}$.

Theorem (Wiles et al.): The function

$$L(C,s) = \prod_p \left(\frac{1}{1 - a_p p^{-s} + p^{1-2s}}\right)$$

extends to an entire complex-analytic function on $\mathbf{C}$.

```
L = EllipticCurve([-2009^2,0])._pari_().elllseries
show(line([(i,L(i)) for i in [0,0.05,..,2]]), figsize=[7,1.5], ymax=10)
```




```
L = EllipticCurve([-1954^2,0])._pari_().elllseries
```

Wiles and Coates ==> curve has no rational points with $y \neq 0$, so 1954 is not the area of a rational right triangle

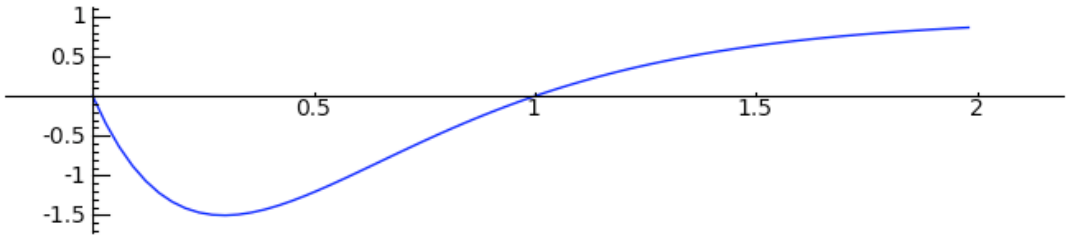
```
L(1)
```

```
1.89814733216426
```

We use the following interact to plot the L-series for many small values of n (e.g., 1,2,3,4,5,6,7,8,9,10). For which $n \leq 10$ does $L(1) = 0$?

```
@interact
def example(n=6):
   L = EllipticCurve([-n^2,0])._pari_().elllseries
   show(line([(i,L(i)) for i in [0,0.03,..,2]]), figsize=[7,1.5])
```

n 6



The Birch and Swinnerton-Dyer Conjecture

Heuristic Observation: If C has infinitely many rational points, then the numbers N_p will tend to be "large". Since $L(C,1)$" = " $\prod_p \frac{p}{N_p}$, the number $L(C,1)$ will tend to be small.

Theorem (Mordell): There is a finite set $P_1, \ldots, P_r$ of rational points on C so that all (non-torsion) rational points can be generated from these using a simple geometric process (chords and tangents).

We call the smallest r in Mordell's theorem the ***rank*** of C.

Conjecture (Birch and Swinnerton-Dyer):

$$\text{ord}_{s=1} L(C, s) = \text{rank}(C)$$

This problem, exactly as stated, is the Clay Math Institute ***Million Dollar prize problem*** in algebraic number theory. Its solution would also resolve the 1000-year old congruent number problem.

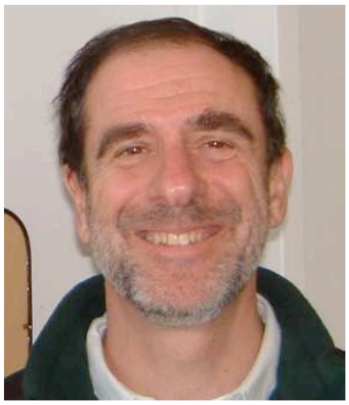
Examples of the BSD Conjecture

```
@interact
def _(n=6):
   x,y = var('x,y')
   C = EllipticCurve(y^2 == x^3 - n^2*x)
   show(C)
   print "rank = ", C.rank(), "\n"
   L = C.lseries()
   print "L-series = ", L.taylor_series(1,53, 4)
```

The Kolyvagin -- Gross-Zagier Theorem

Theorem: *If* $\text{ord}_{s=1} L(C, s) \leq 1$ *then the Birch and Swinnerton-Dyer conjecture is true for* C.

The proof involves Heegner points, modular curves, Euler systems and Galois cohomology.


My Current Research

- Study the mathematical structures (Heegner points, modular curves, Euler systems, etc.) that appear in the proof of the Kolyvagin-Gross-Zagier theorem in order to understand how to generalize *anything* to cubic curves with $\mathrm{ord}_{s=1} L(C, s) \geq 2$.

- This involves a combination of *technical theory* and *explicit machine computation*.