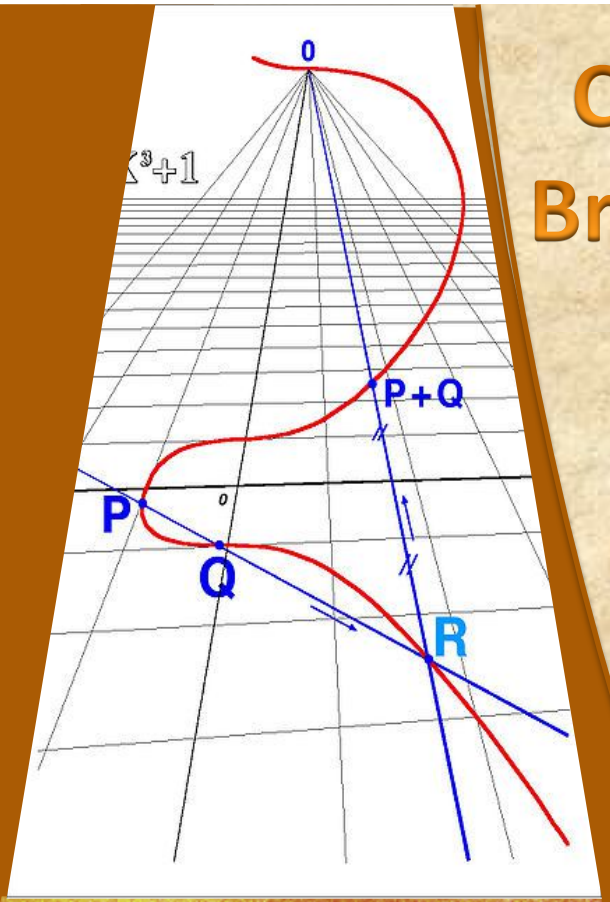
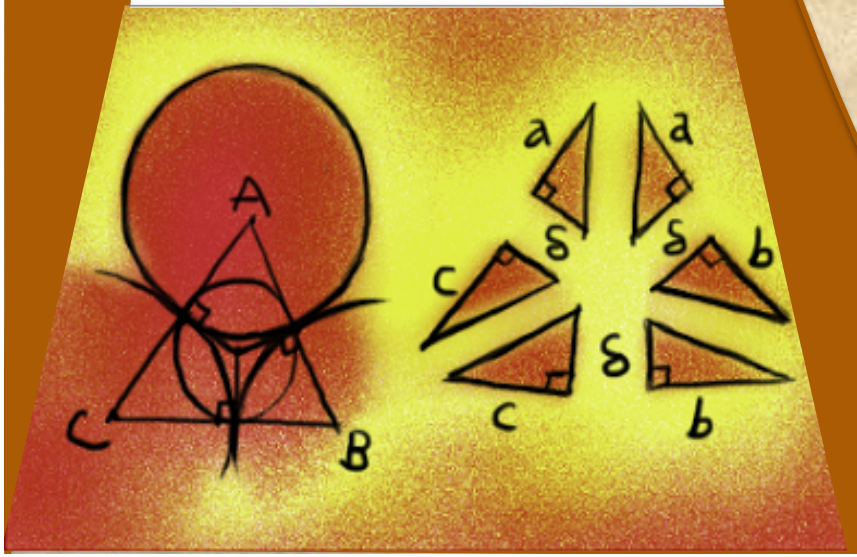




Curiosando nella Matematica
Browsing through Mathematics
Torino, 28-04-2014

From the Heron formula to elliptic curves

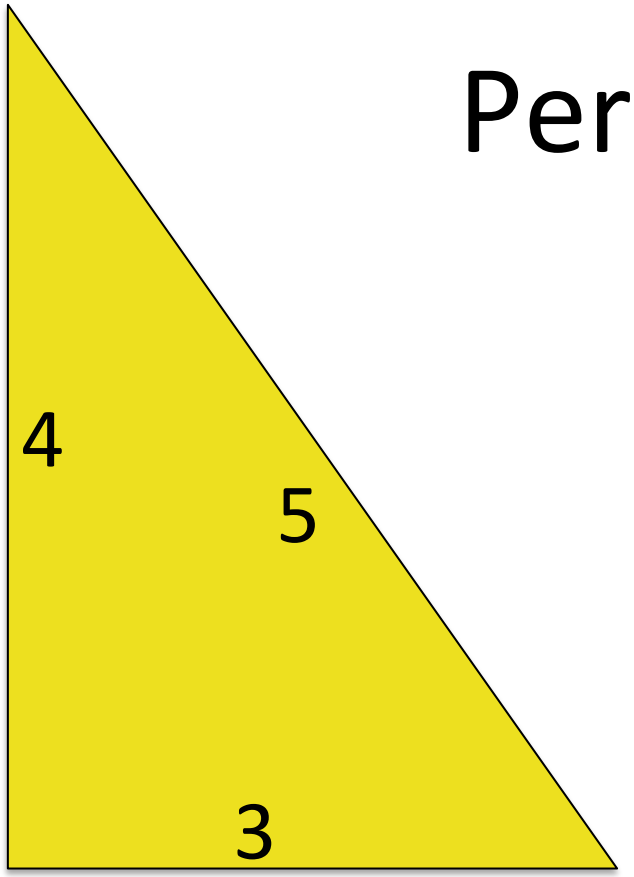
Ferdinando Arzarello

The story I will tell you in this presentation aims at showing the remarkable unity of mathematics, starting as it does in high school and ending in research.

Overview

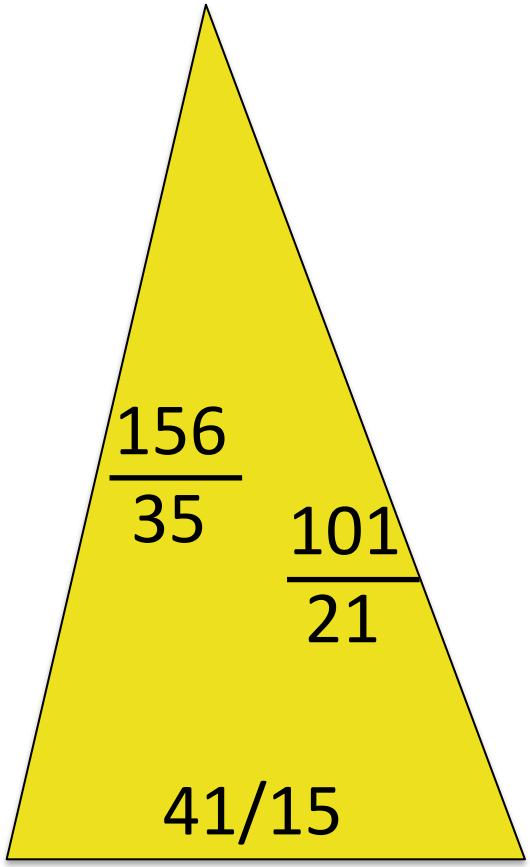
1. A problem of triangles
2. Elliptic curves
3. Factoring big numbers
4. Elliptic cryptography

A secondary school problem:

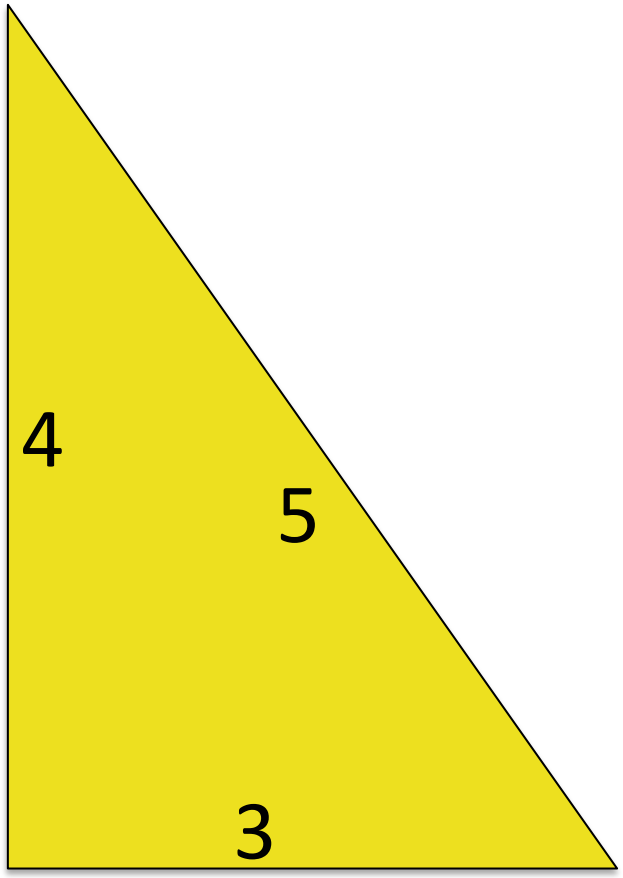


Perimeter = ?

Area = ?



A very easy secondary school problem:



Semiperimeter = 6

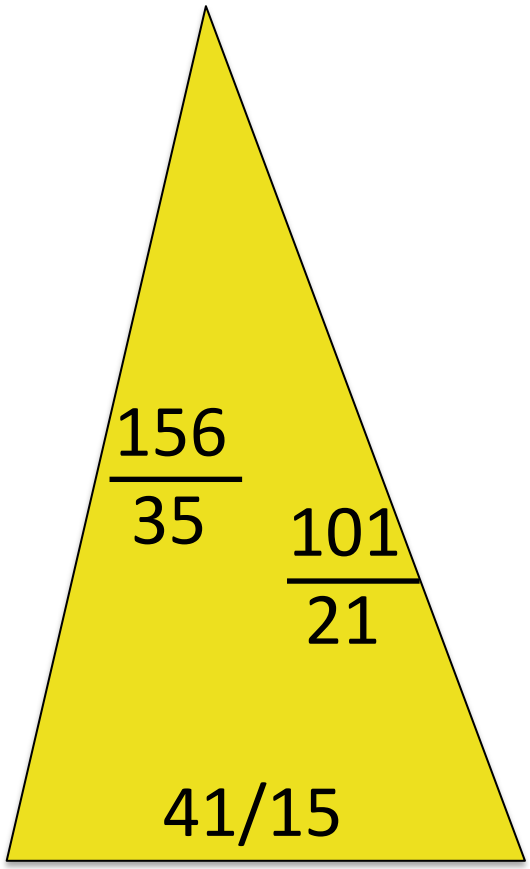
Area = 6

Using Heron formula…

$$\text{Area } \Delta ABC = \sqrt{s(s-a)(s-b)(s-c)}$$

Semiperimeter $s = 6$

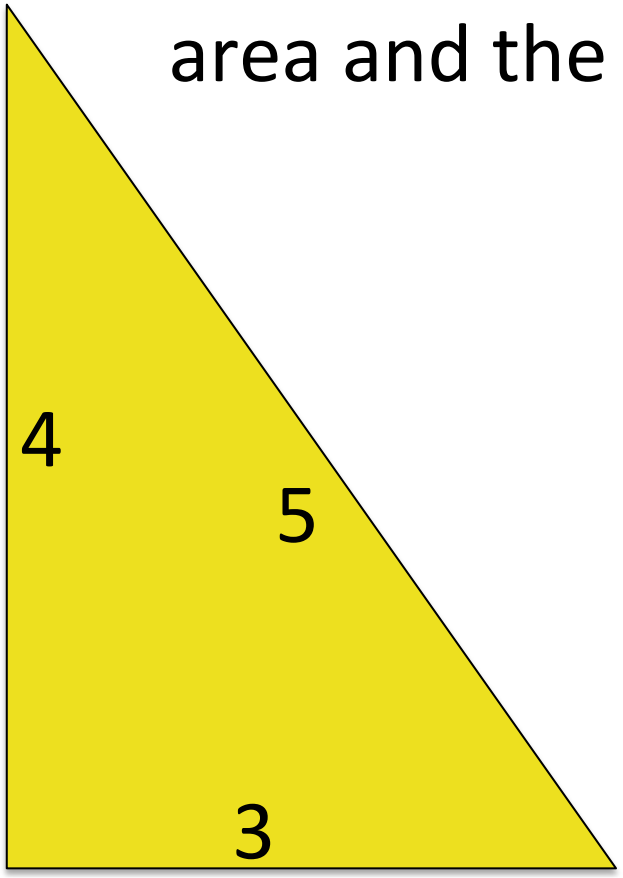
Area = 6

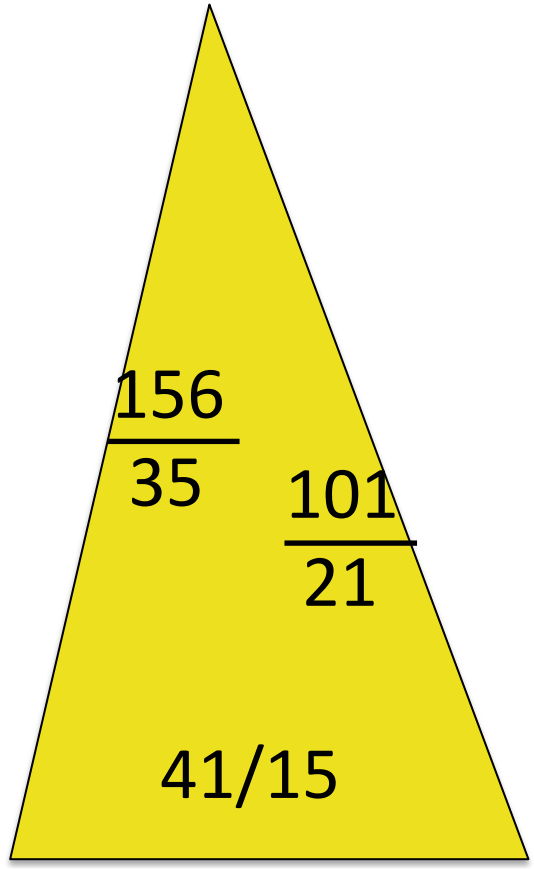


A second problem, not so easy

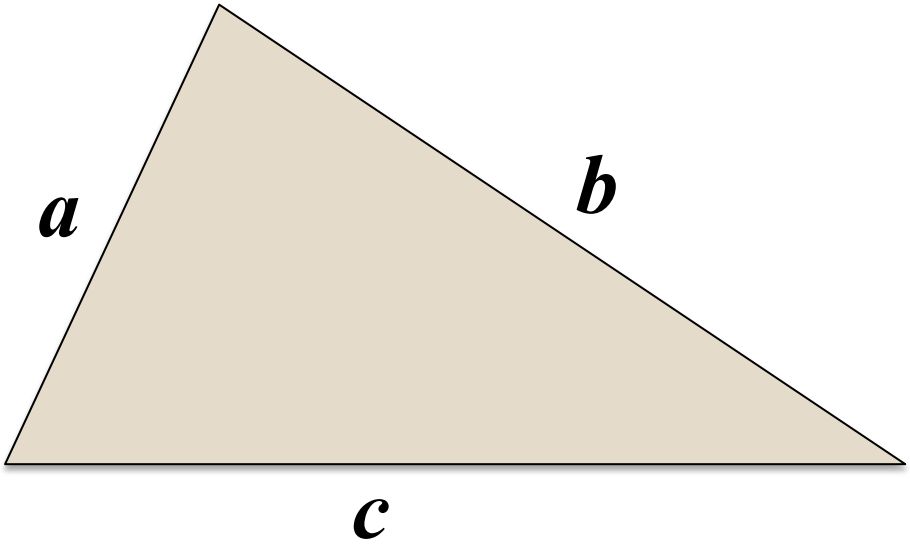
The two triangles have the same area AND the same perimeter.

How to find ALL triangles having the same area and the same perimeter?

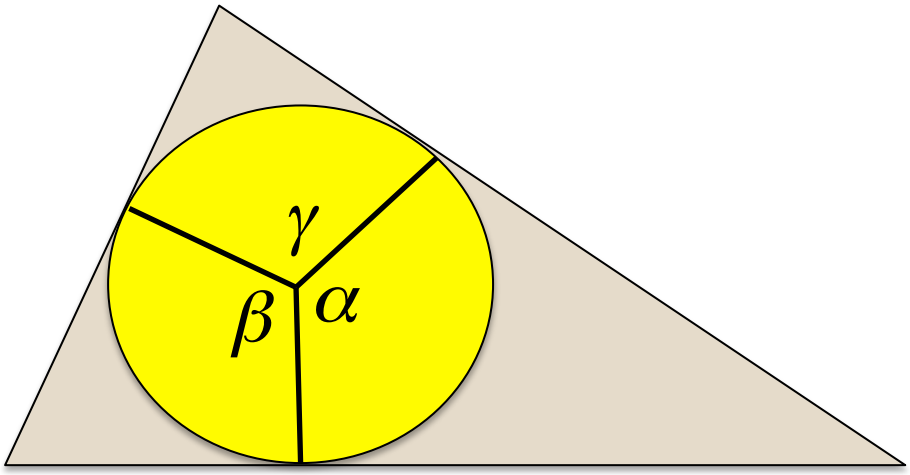




Sides and angles

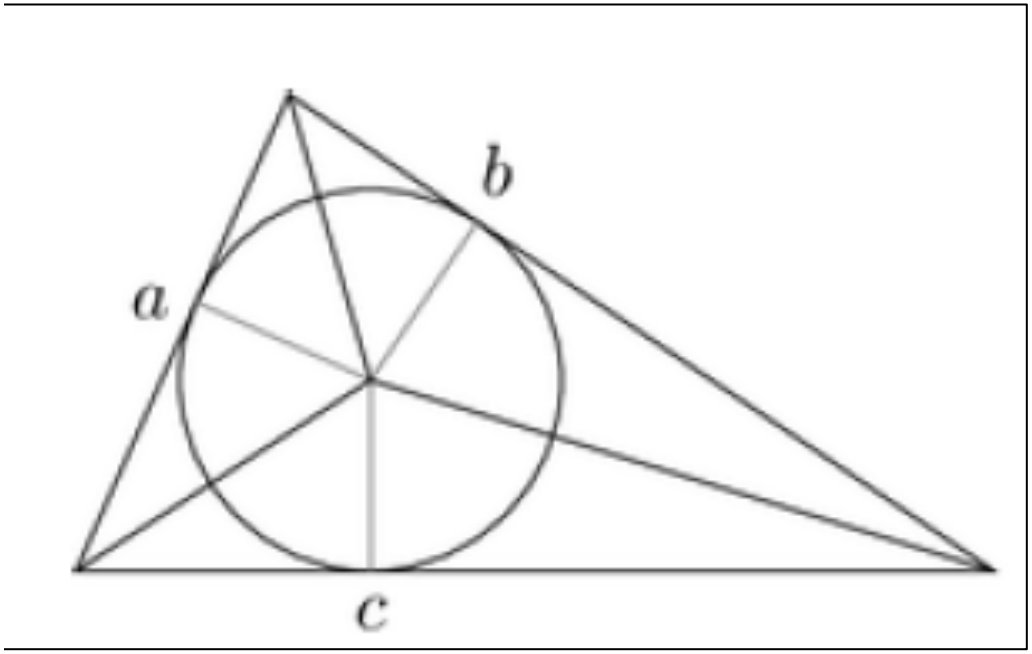


a, b, c



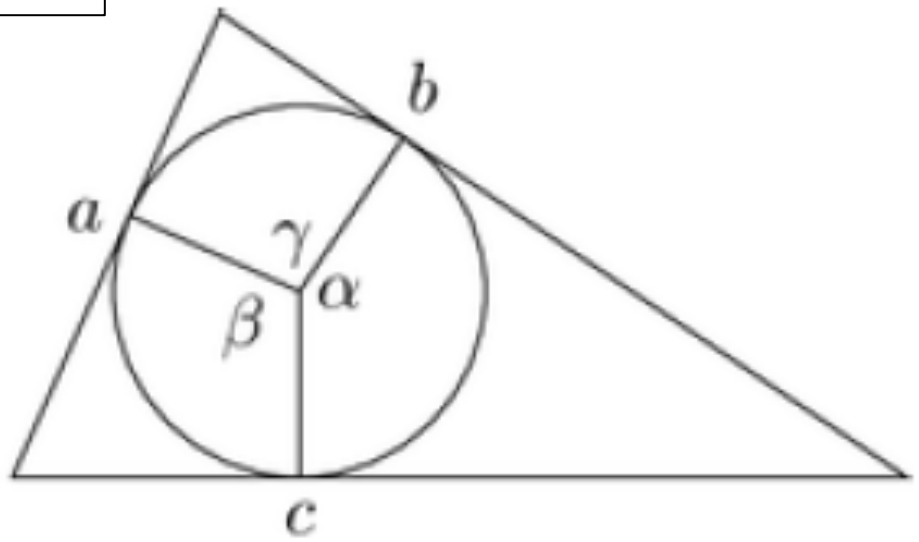
α, β, γ

Sides and angles

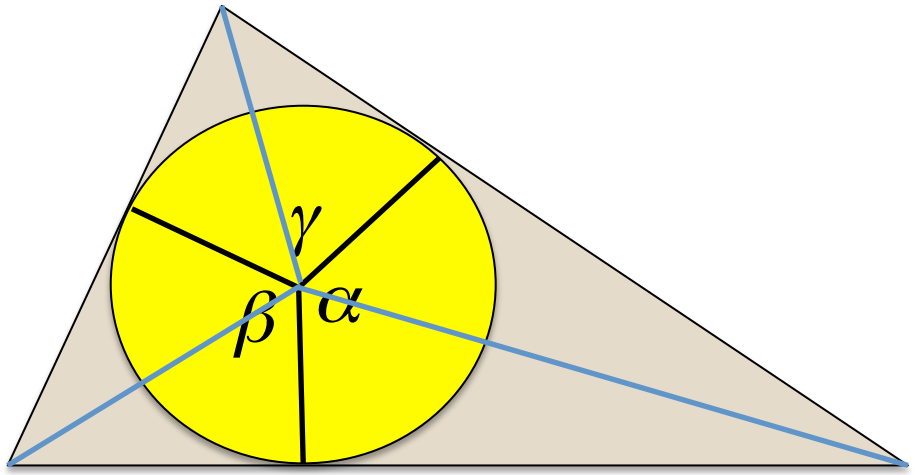


a, b, c

α, β, γ



Sides and angles



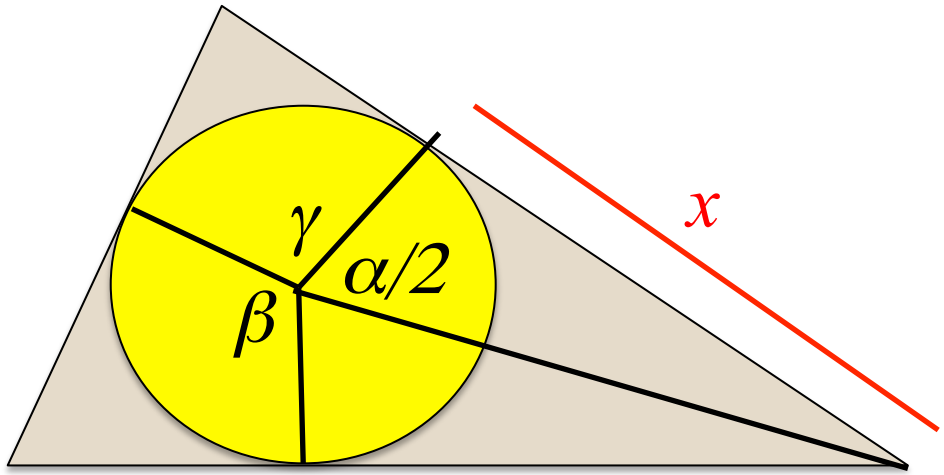
$A = area$

$r = radius$

$s = semiper.$

$$A = rs$$

Sides and angles



$A = area$

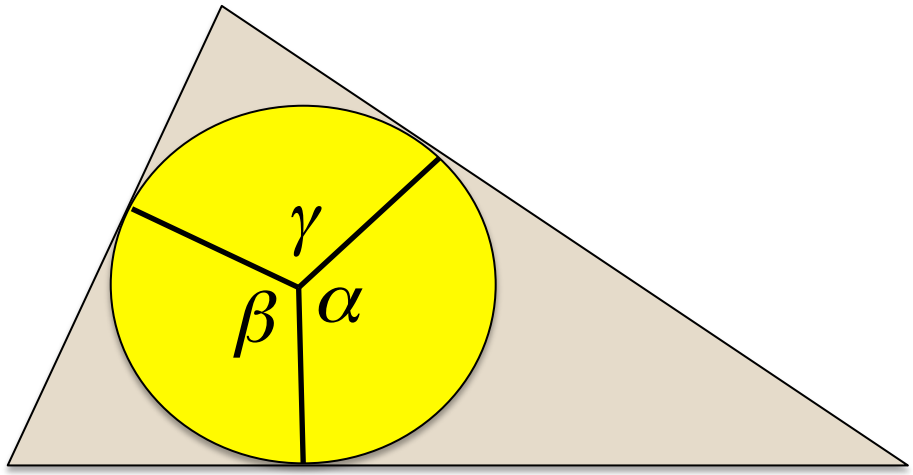
$r = radius$

$s = semiper.$

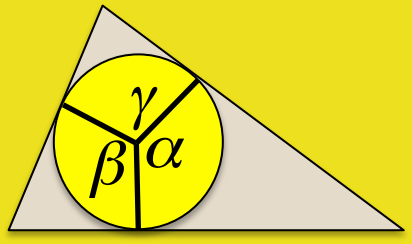
$$x = r \tan(\alpha/2)$$

$$s = r\left(\tan\frac{\alpha}{2} + \tan\frac{\beta}{2} + \tan\frac{\gamma}{2}\right)$$

Sides and angles



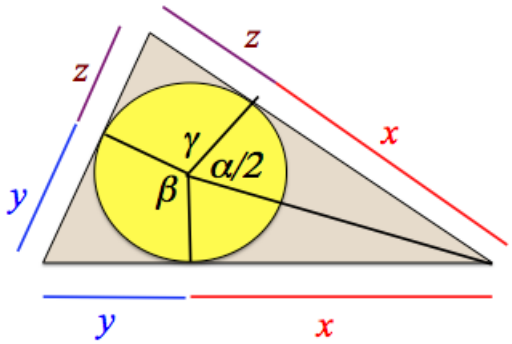
$$\begin{cases} A = rs \\ s = r\left(\tan\dfrac{\alpha}{2} + \tan\dfrac{\beta}{2} + \tan\dfrac{\gamma}{2}\right) \end{cases}$$



Some calculations

$$\begin{cases} A = rs \\ s = r\left(\tan\frac{\alpha}{2} + \tan\frac{\beta}{2} + \tan\frac{\gamma}{2}\right) \end{cases}$$

$$\tan\frac{\alpha}{2} + \tan\frac{\beta}{2} + \tan\frac{\gamma}{2} = \frac{s^2}{A}.$$



$$\frac{\gamma}{2} = \pi - \frac{\alpha}{2} - \frac{\beta}{2}$$

$$z = \tan\left(\frac{\gamma}{2}\right) = \tan\left(\pi - \frac{\alpha}{2} - \frac{\beta}{2}\right) = -\tan\left(\frac{\alpha}{2} + \frac{\beta}{2}\right) = -\frac{x+y}{1-xy}$$

$$x + y - \frac{x+y}{1-xy} = k = \frac{s^2}{A}$$

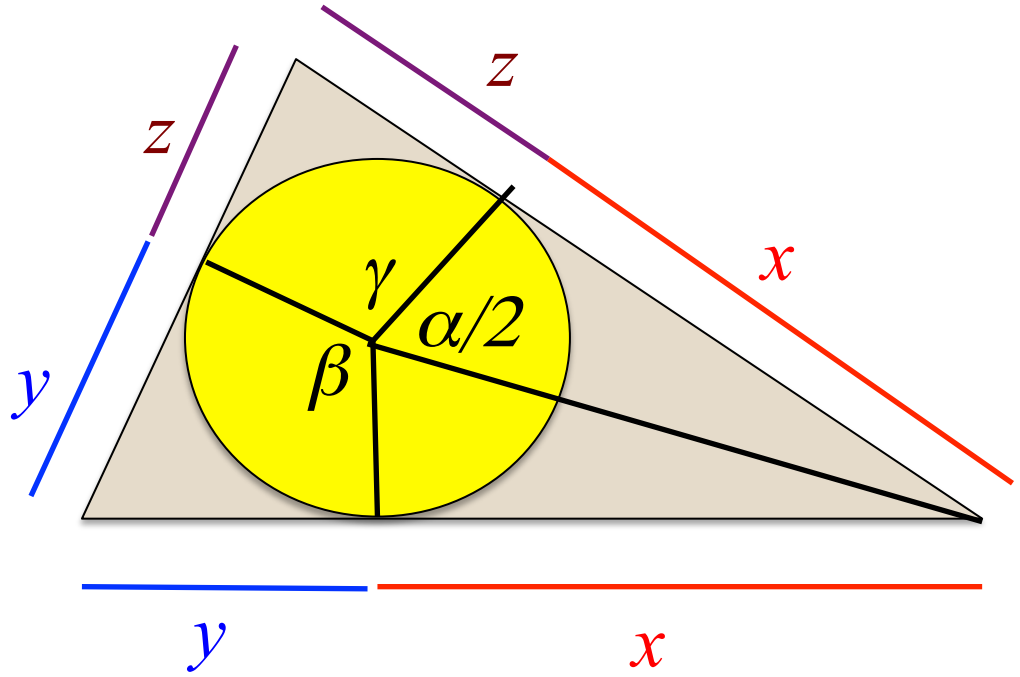
$$x^2y + xy^2 = kxy - k \qquad x > 0,\ y > 0,\ xy > 1$$

Some constrains

$$x^2y + xy^2 - kxy + k = 0$$

$$x > 0,\ y > 0,\ xy > 1\ ,\ k = s^2/A > 0$$

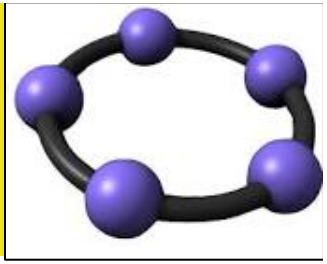
Reading the results



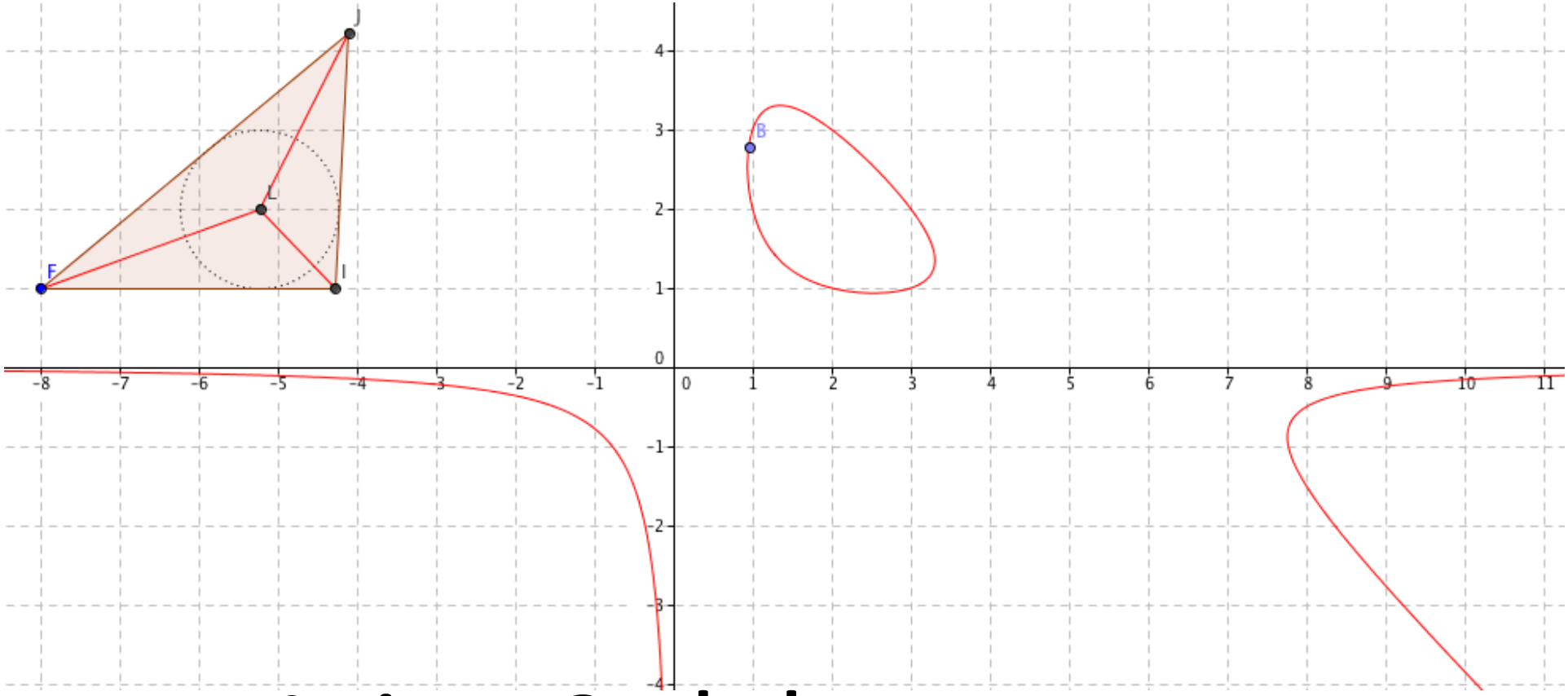
$r = radius$

$x = r \tan(\alpha/2)$ $y = r \tan(\beta/2)$ $z = r \tan(\gamma/2)$

In short…



$$x^2y + xy^2 - kxy + k = 0$$



It is a 3rd degree curve

Overview

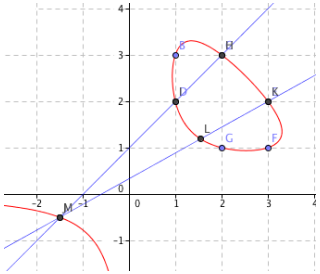
1. A problem of triangles
2. **Elliptic curves**
3. Factoring big numbers
4. Elliptic cryptography

From Heron to many old and current research problems

- Elliptic curves, integrals, and functions

$$L = 4a \int_0^{\pi/2} \sqrt{1 - k^2 \sin^2 t}\, dt.$$

- Elliptic curves and rational points



- Elliptic curves and the proof of the Last Fermat theorem

OBSERVATIO DOMINI PETRI DE FERMAT.

CVbum autem in duos cubos, aut quadratoquadratum in duos quadratoquadratos & generaliter nullam in infinitum vltra quadratum poteſtatem in duos eiuſdem nominis fas eſt diuidere cuius rei demonſtrationem mirabilem ſane detexi. Hanc marginis exignitas non caperet.

- Elliptic curves factorization algorithms and cryptography

Security level (in bits)	Symmetric scheme (key size in bits)	ECC-based scheme (size of n in bits)	DSA/RSA (modulus size in bits)
56	56	112	512
80	80	160	1024
112	112	224	2048
128	128	256	3072
192	192	384	7680
256	256	512	15360

The last Fermat theorem

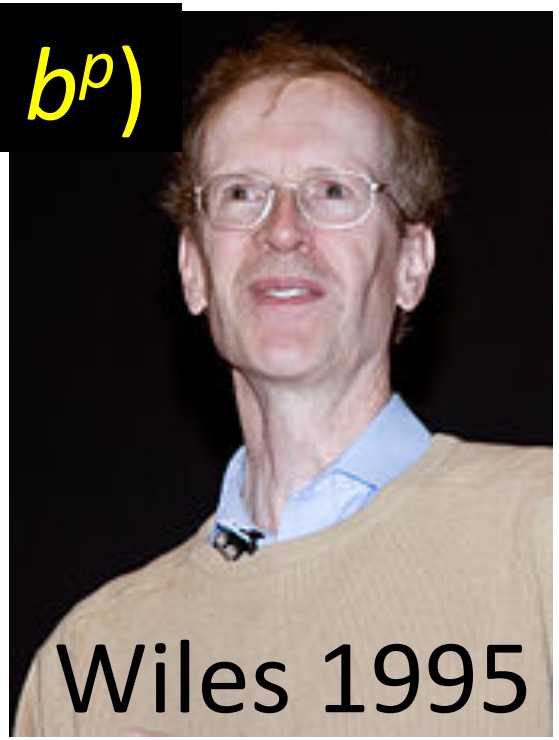
OBSERVATIO DOMINI PETRI DE FERMAT.

Cubum autem in duos cubos, aut quadratoquadratum in duos quadratoquadratos & generaliter nullam in infinitum vltra quadratum poteſtatem in duos eiuſdem nominis fas eſt diuidere cuius rei demonſtrationem mirabilem ſane detexi. Hanc marginis exiguitas non caperet.

Fermat 1637

$$y^2 = x\,(x - a^p)(x + b^p)$$

G. Frey 1984



Wiles 1995

Integrals and elliptic functions

$$L_a^b = \int_a^b \sqrt{1 + (f'(x))^2}\, dx \qquad L = 4\int_0^a \frac{\sqrt{a^2 - k^2x^2}}{\sqrt{a^2 - x^2}}\, dx$$

$$k = \sqrt{a^2 - b^2}/a$$

$$L = 4a\int_0^{\pi/2} \sqrt{1 - k^2 \sin^2 t}\, dt.$$

ECC Vs/ RSA

Security level (in bits)	Symmetric scheme (key size in bits)	ECC-based scheme (size of n in bits)	DSA/RSA (modulus size in bits)
56	56	112	512
80	80	160	1024
112	112	224	2048
128	128	256	3072
192	192	384	7680
256	256	512	15360

Neal Kobliz, Victor Miller

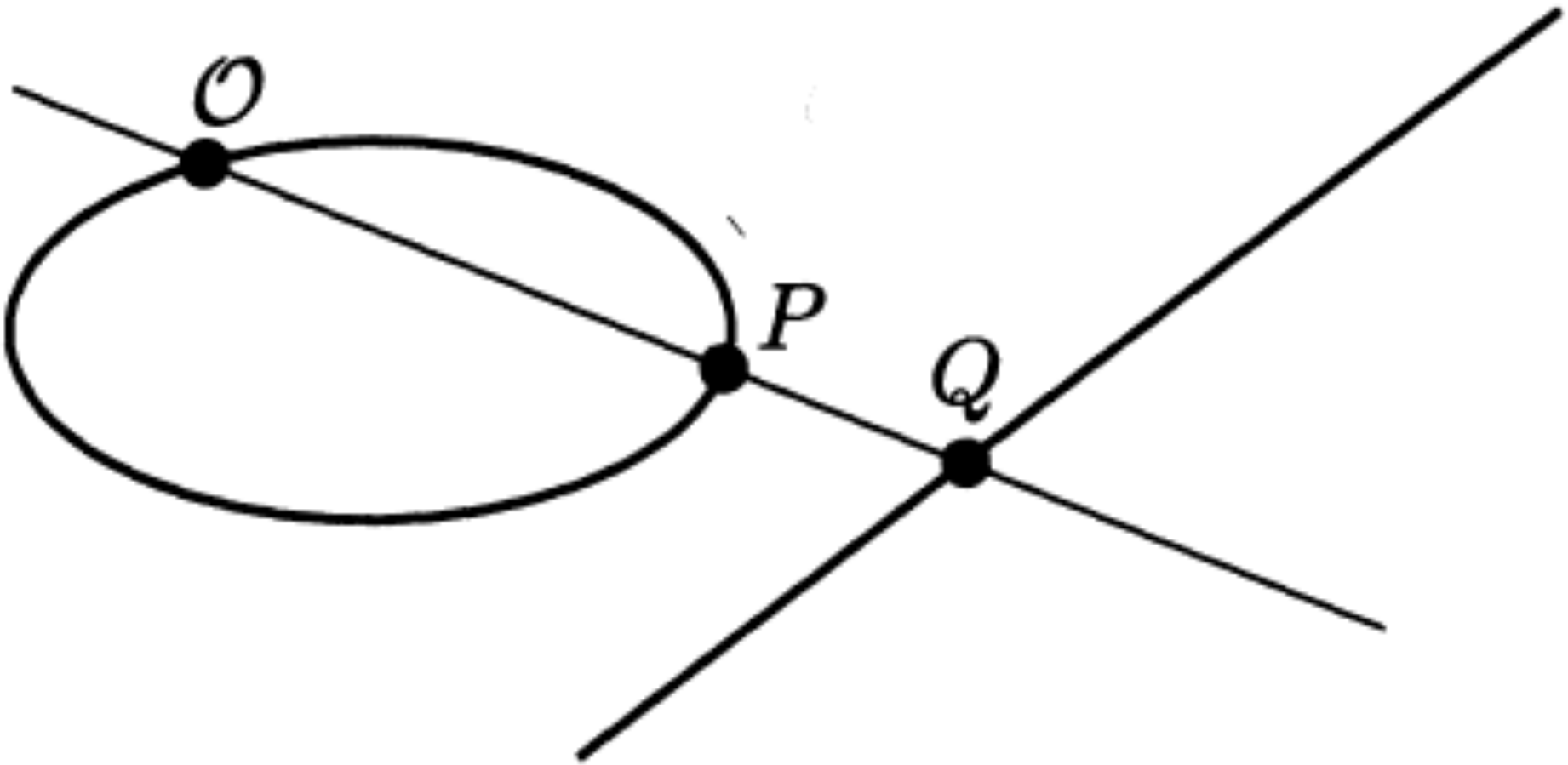
R. Rivest, A. Shamir, L. Adleman

From Heron to many old and current research problems

- Elliptic curves, integrals, and functions
- Elliptic curves and rational points
- Elliptic curves and the proof of the Last Fermat theorem
- Elliptic curves factorization algorithms and cryptography

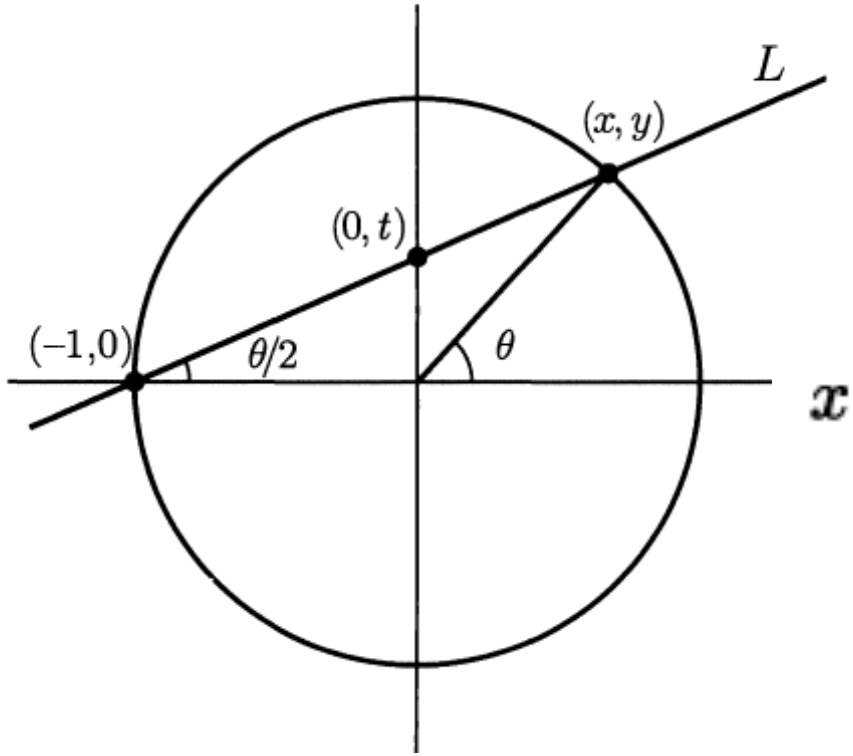
Rational points on curves

The rational points on a conic are in one-to-one correspondence with the rational points on a line.



Let's carry out this procedure for the circle

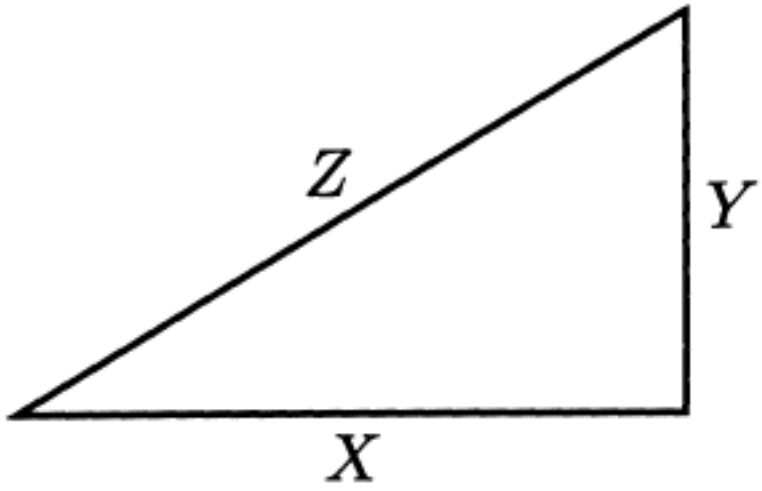
$$x^2+y^2 = 1$$



$$x = \frac{1-t^2}{1+t^2}, \qquad y = \frac{2t}{1+t^2}.$$

If x and y are rational numbers, then t will be a rational number.

In particular, the point
(x, y) defined by
$x = X/Z$, $y = Y/Z$
is a rational point on the
circle $x^2 + y^2 = 1$



(the rational numbers x and y are in lowest terms).

Since (x,y) is a rational point on the circle, there is some rational number t so that x and y are given by the formulas we derived above. Write $t = m/n$ in lowest terms. Then:

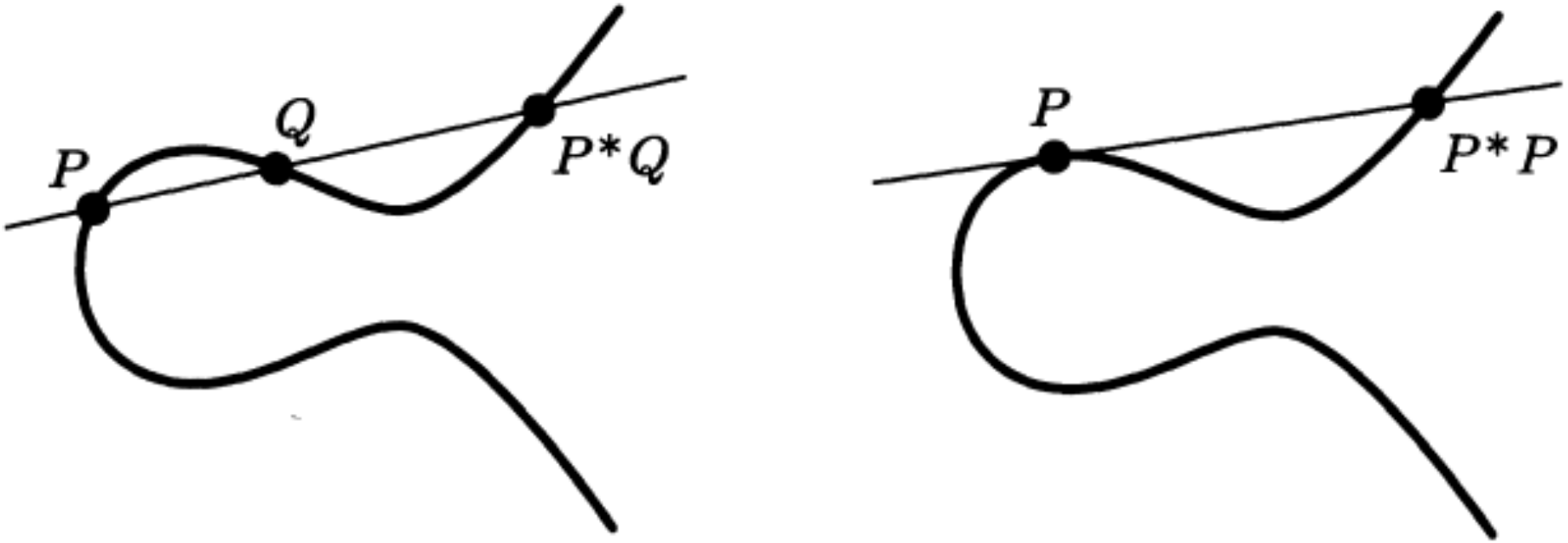
$$\frac{X}{Z} = x = \frac{n^2 - m^2}{n^2 + m^2}, \qquad \frac{Y}{Z} = y = \frac{2mn}{n^2 + m^2}.$$

This proves that to get all primitive triangles, you take two relatively prime integers m and n and let:

$$X = n^2 - m^2$$
$$Y = 2mn$$
$$Z = n^2 + m^2$$

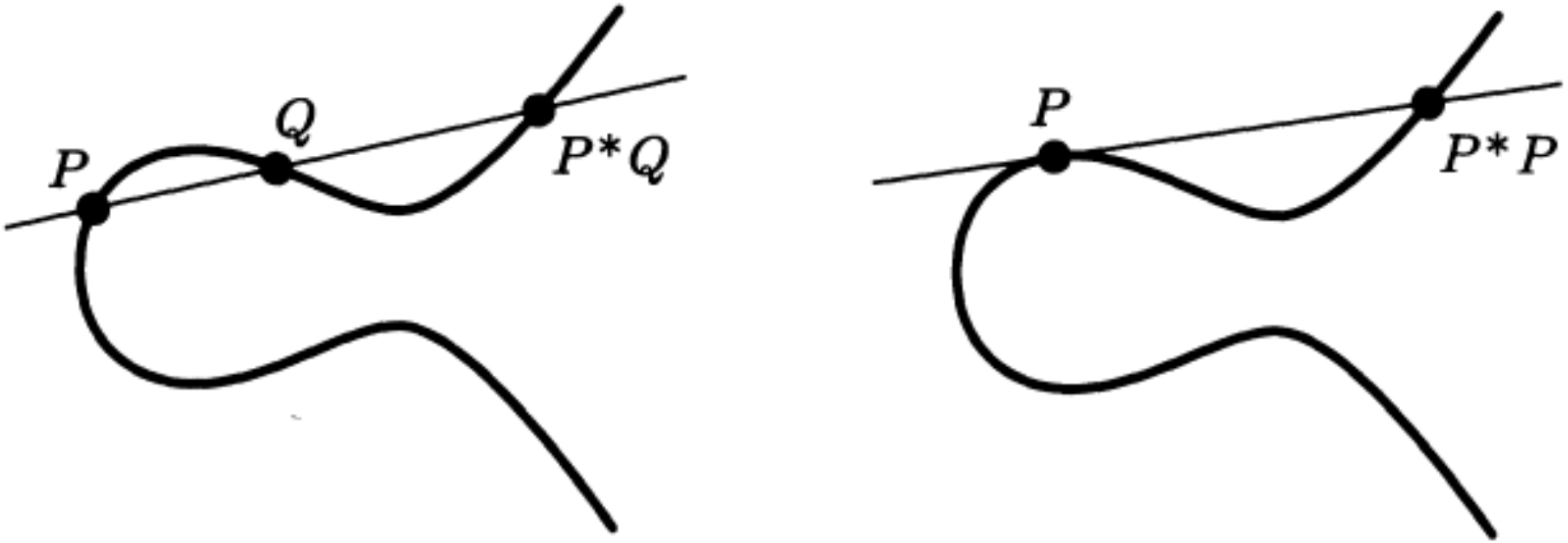
(X odd, Y even)

And the cubics?

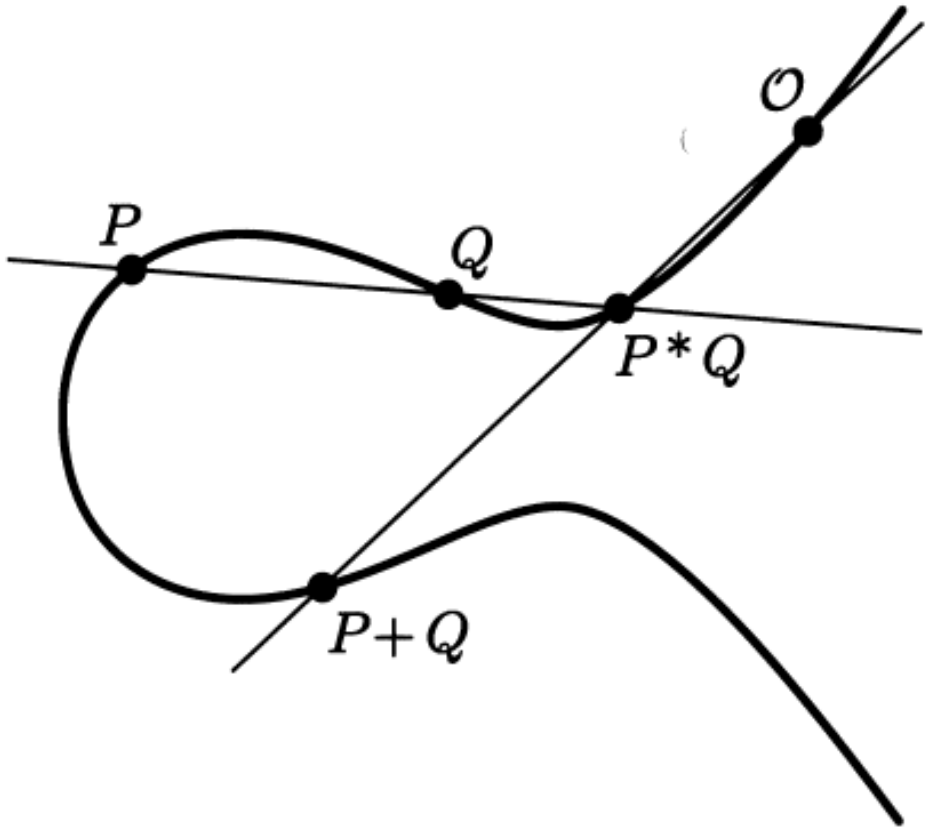


If we can find two rational points on the curve, then we can generally find a third one, provided its coefficients are rational.

And the cubics?

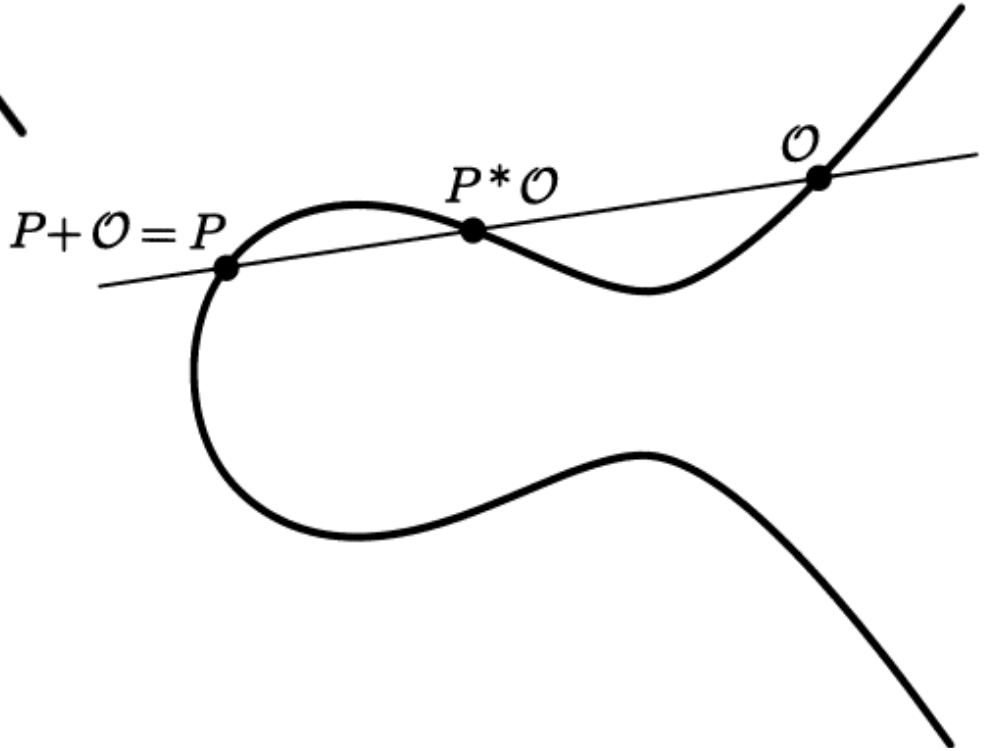


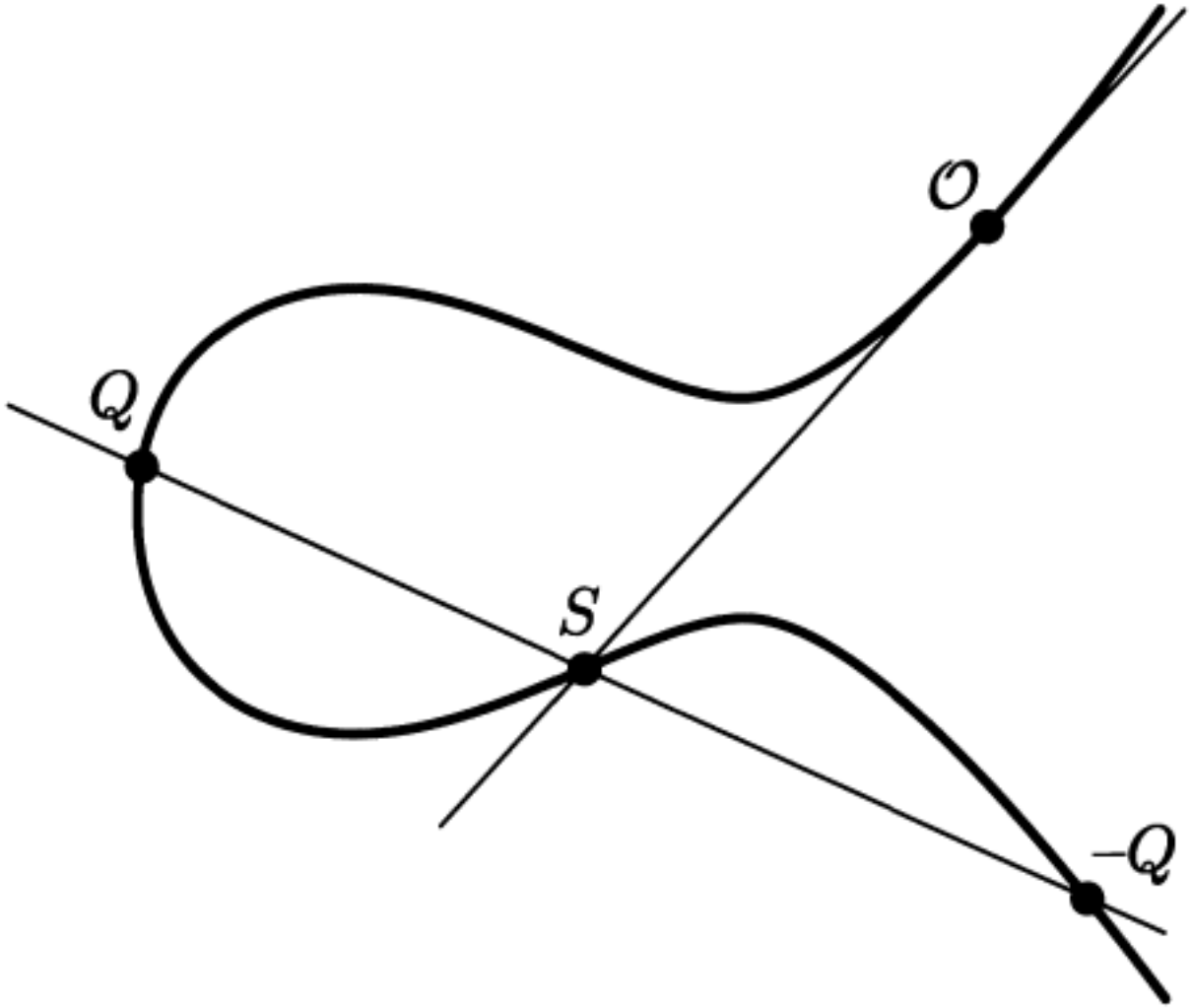
A theorem of Mordell states that if C is a non-singular rational cubic curve, then there is a finite set of rational points such that **all** other rational points can be obtained in this way.



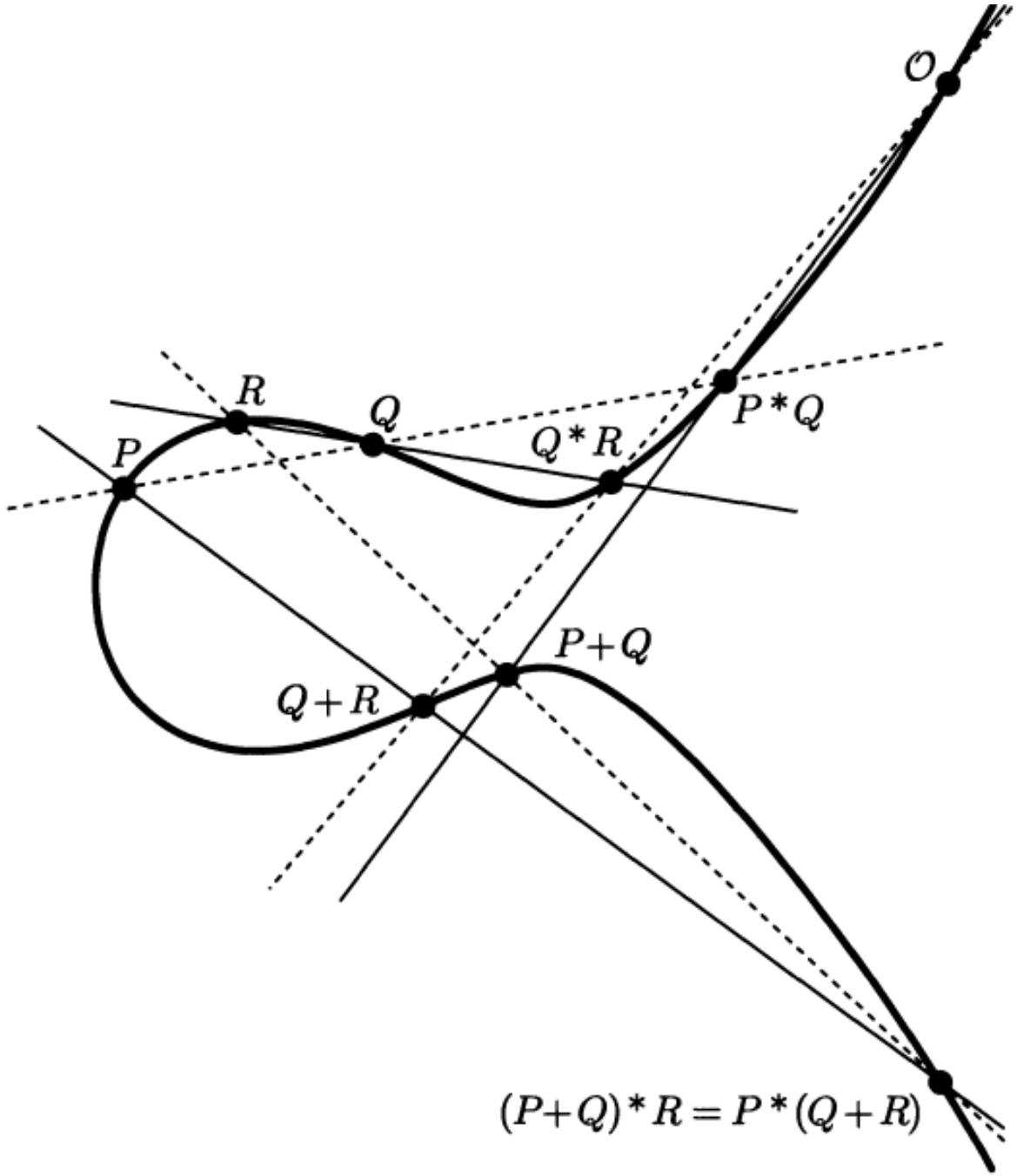
These points can be structured as a commutative group.

We can make it in such a way that a given rational point O becomes the zero element of the group.

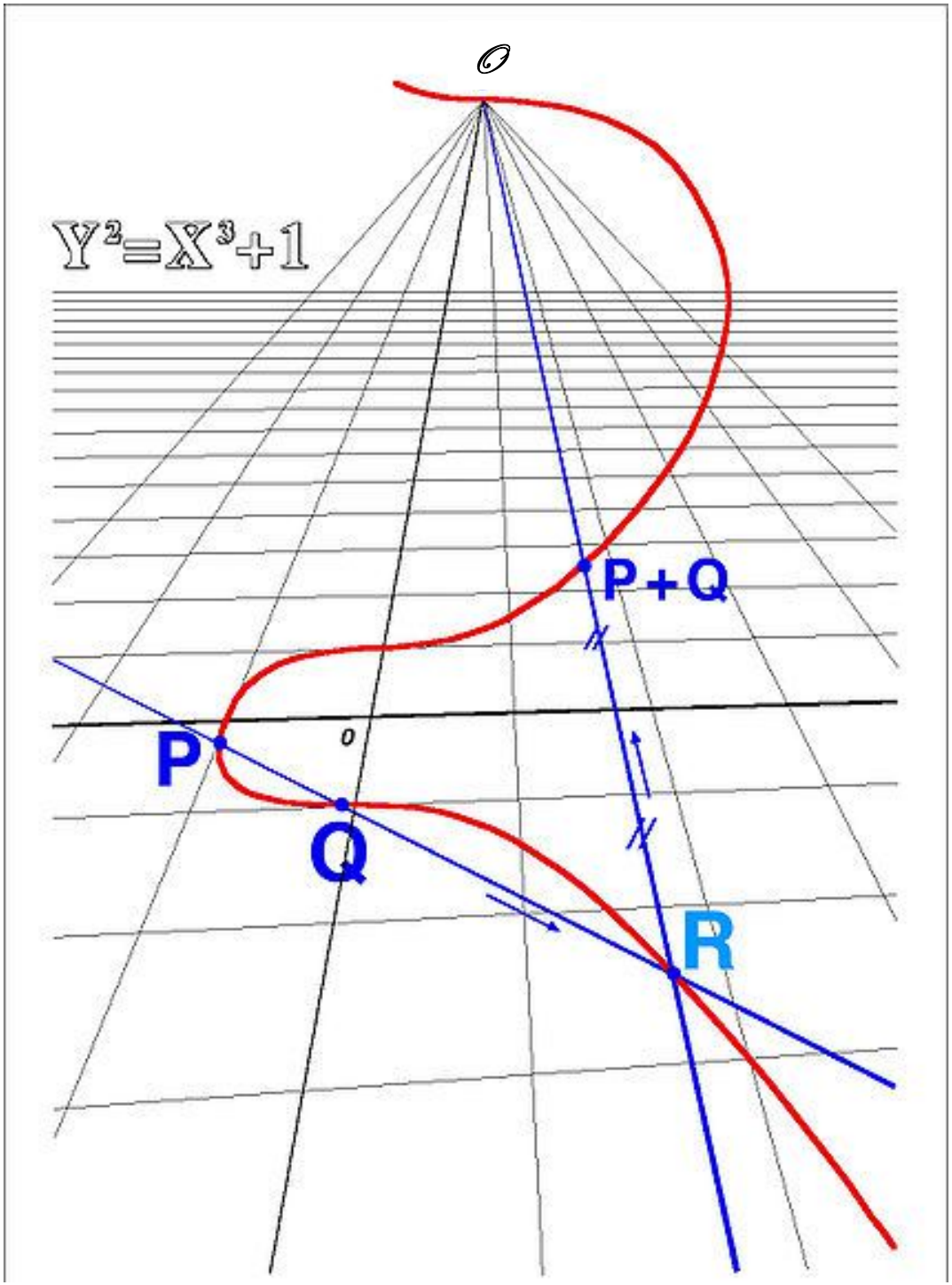




Calculating the opposite of a point Q



Verifying the Associative Law



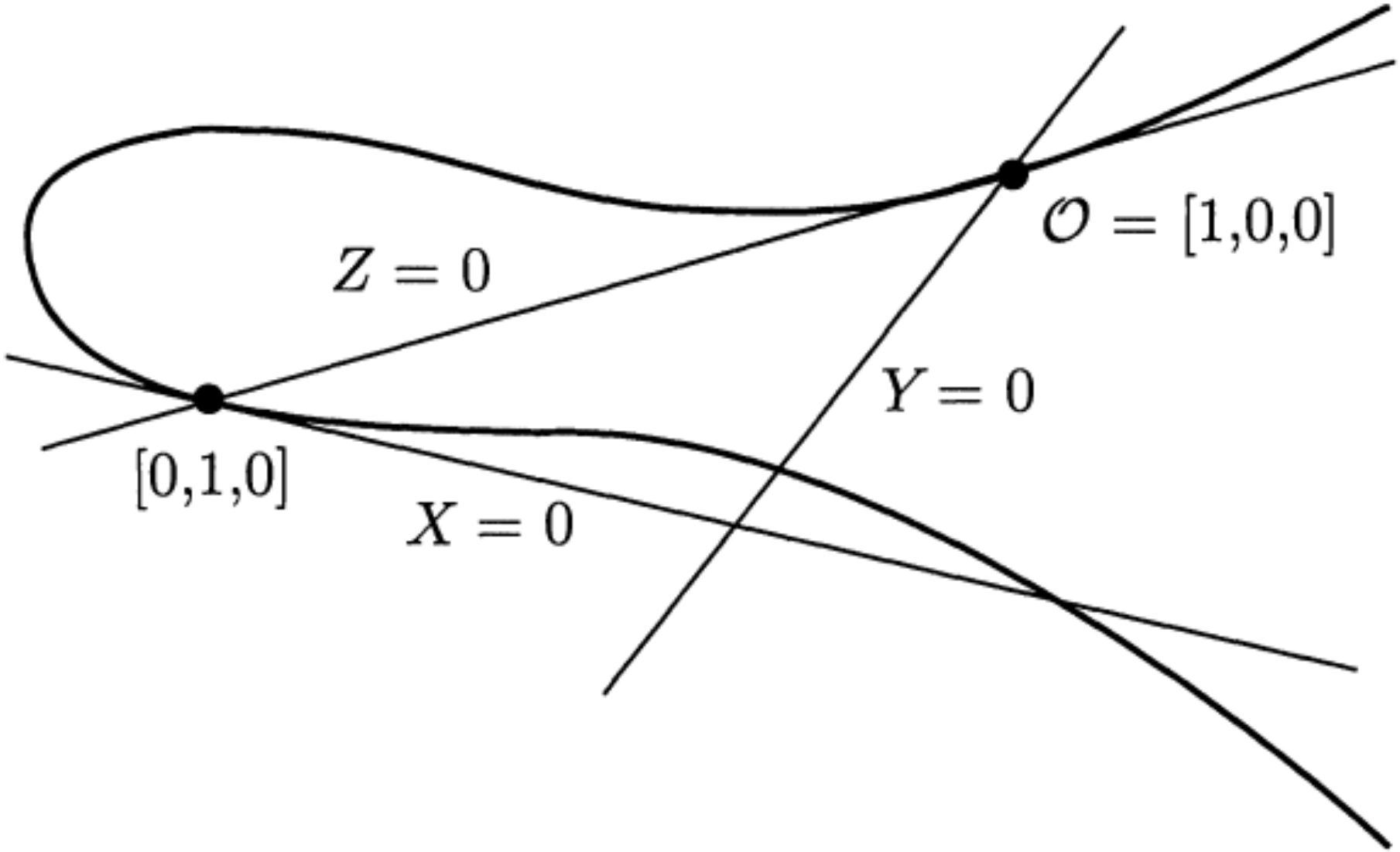
Pushing $\mathcal{O}$ to infinite

Mordell's Theorem.
(1923)

If a non-singular plane cubic curve has a rational point, then the group of rational points is finitely generated.

Mordell Conjecture (For Plane Curves).
*Let C be a smooth rational plane curve of degree strictly greater than 3. Then the set C(*Q*) of rational points on C is finite.*

This conjecture was proved by Faltings in 1983 and is a major achievement in diophantine geometry to which many mathematicians have contributed.

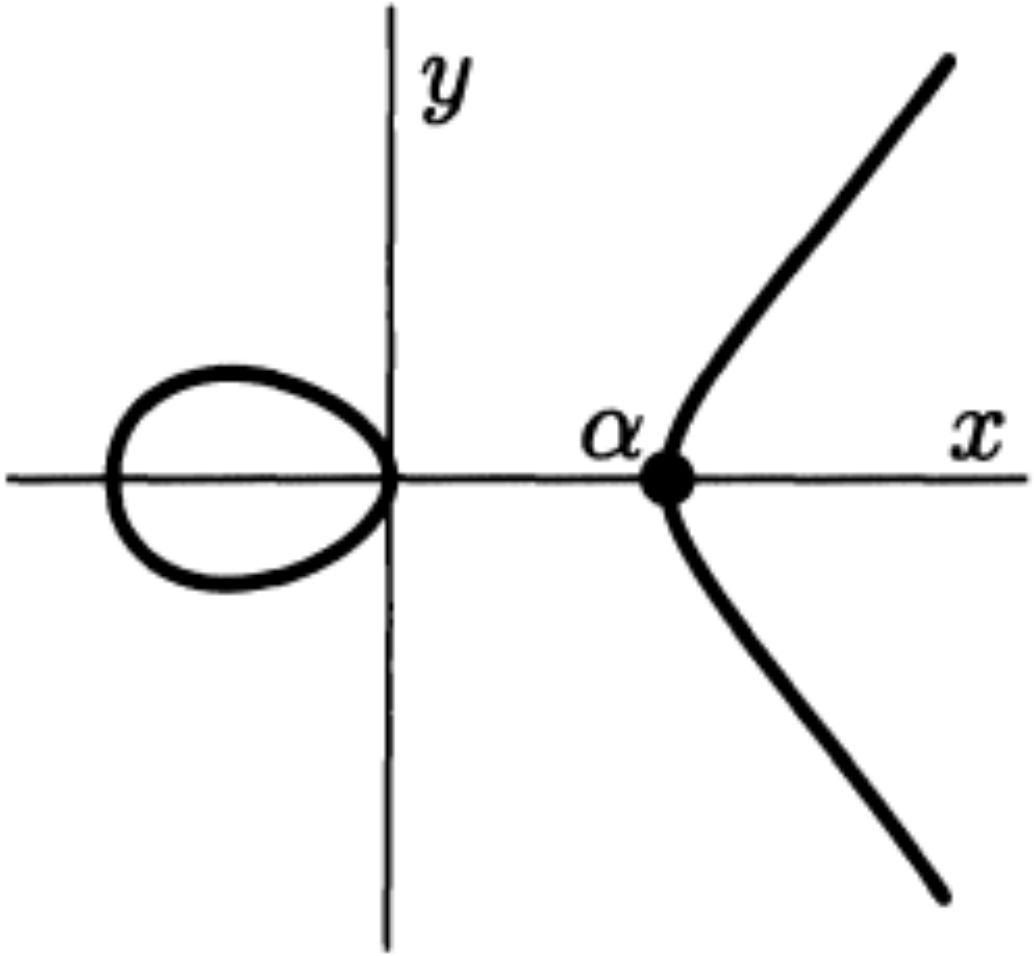


Choosing Axes to Put C into Weierstrass Form

A cubic equation in normal form (of Weierstrass) has equation:

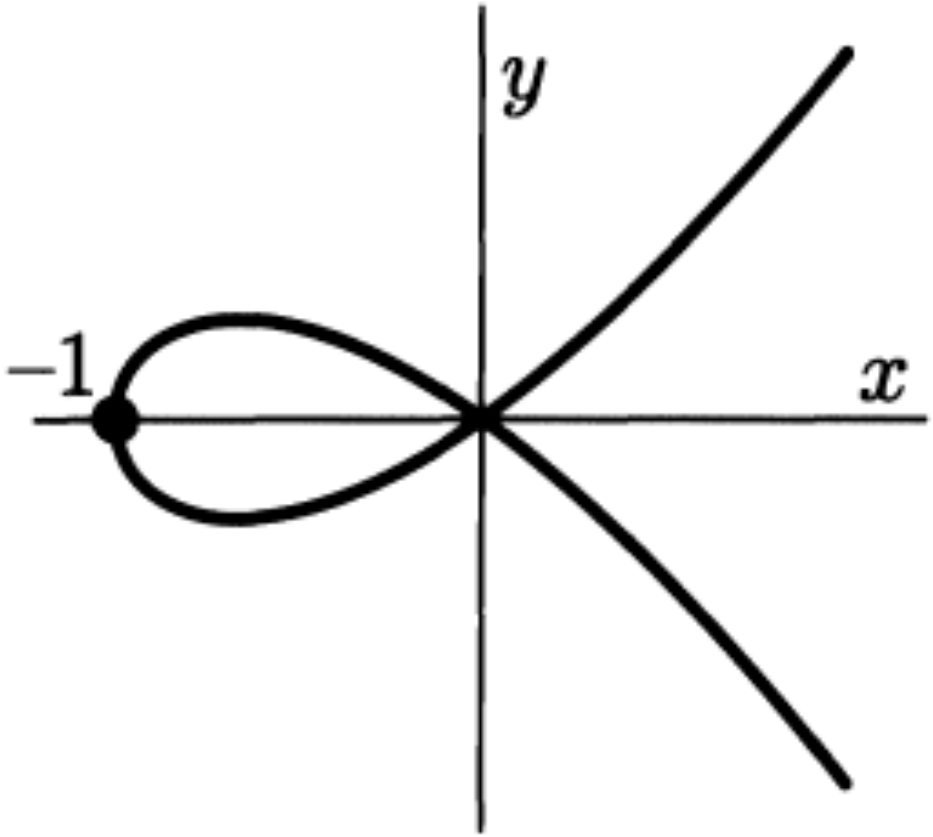
$$y^2 = f(x) = x^3 + ax^2 + bx + c$$

If the (complex) roots are distinct, it is called an *elliptic curve*.

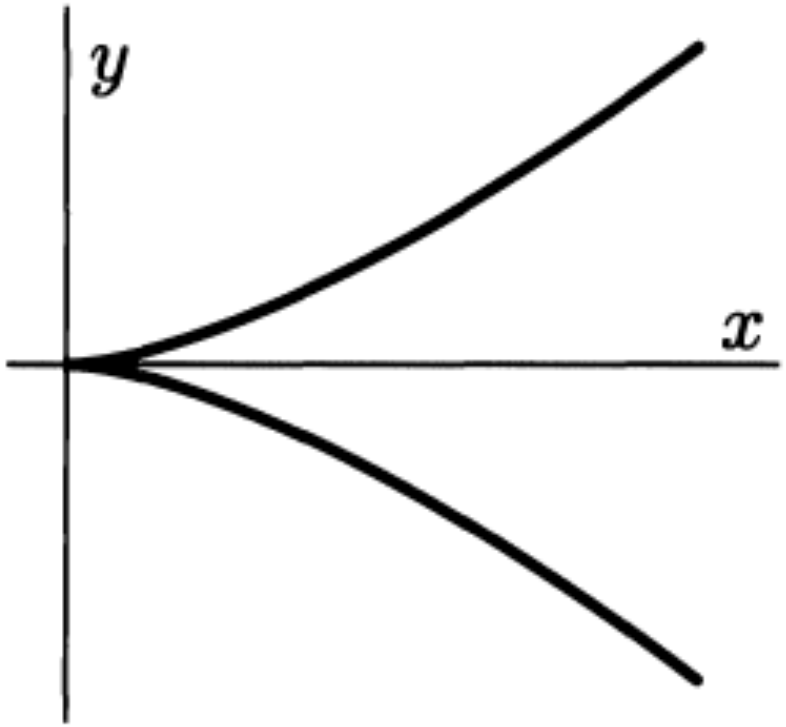


3 distinct real roots (2 components)

Singular cubics



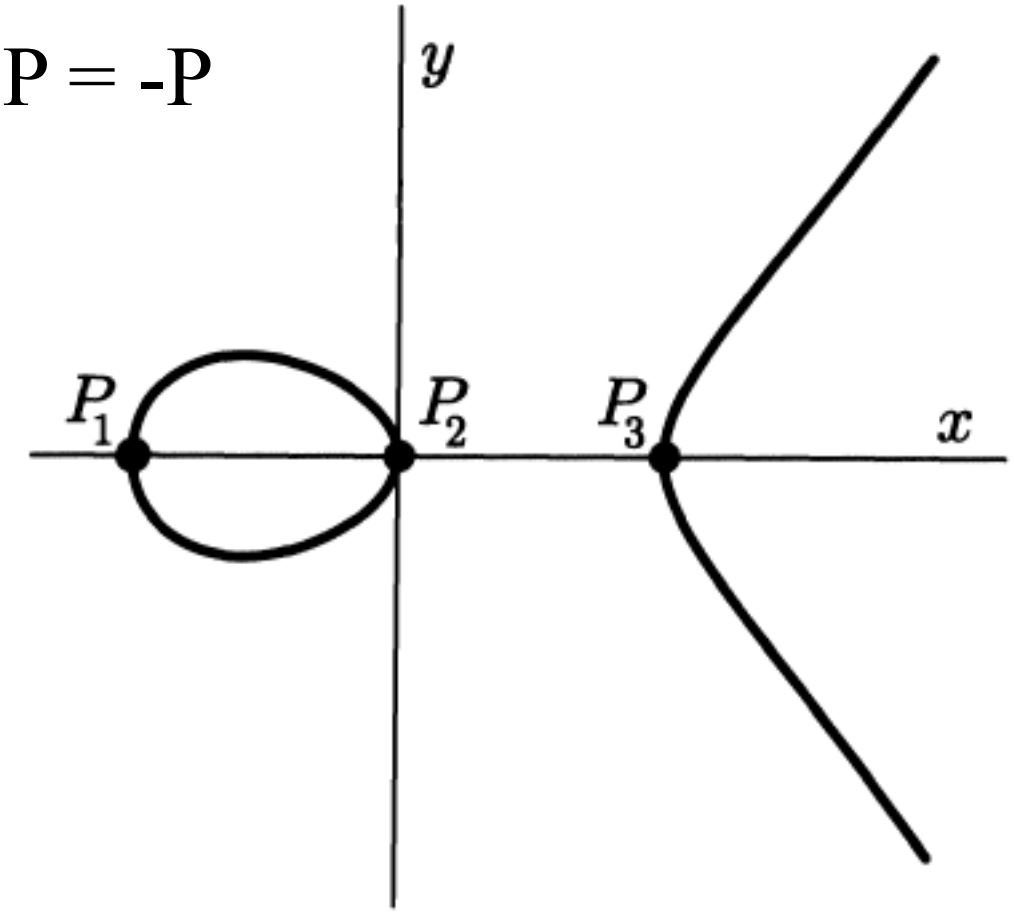
$y^2 = x^2(x + 1)$



$y^2 = x^3$

Order of the element of a group.

$2P = O$ iff $P = -P$



Points of Order Two

The 4-group: $\{ O, P_1, P_2, P_3 \}$

Order 3

C has exactly 9 points of order dividing 3. They form a group, which is a product of two cyclic groups of order 3.

The points of order 3 can also be described as the inflection points of the elliptic curve.

Nagel- Lutz Theorem

Let $y^2 = f(x) = x^3 + ax^2 + bx + c$
be a non-singular cubic curve with integer coefficients a, b, c; and let D be the discriminant of $f(x)$,

$$D = -4a^3c + a^2b^2 + 18abc - 4b^3 - 27c^2$$

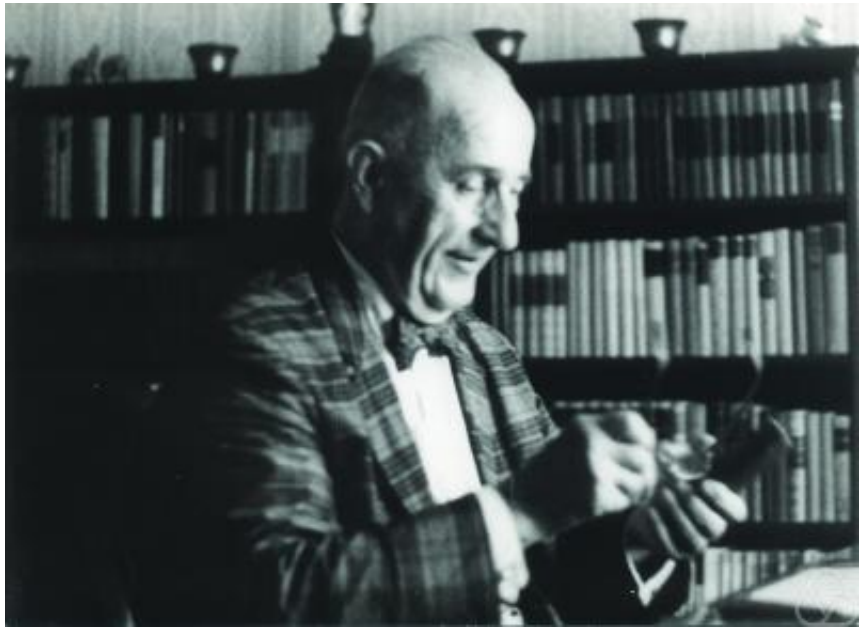
Let P = (x, y) be a rational point of finite order. Then x and y are integers and either $y = 0$, in which case P has order two, or else y divides D.

Follow ups after Nagel-Lutz theorem

In case the curve has rank zero, C(Q) is finite, hence by NL all its rational points are integer.

If the rank is positive, the **Siegel theorem** clarifies that not many points are integer:

The number of integral points on a nonsingular curve of genus strictly greater than 0 and defined over the rational numbers is finite.



Diophantine approximations

Finiteness of solutions in integers of C illustrates the impossibility of nicely approximating algebraic numbers with rationals.

There is a list of results on such a subject, which date to 1909 with the Thue theorem, include the result of Siegel (1921) and of Gelfond and Dyson (1947), to arrive to the best possible result of Roth in 1955.

Theorem of Thue. *Let b the root of an irreducible polynomial f(x) ε Q[X] having degree d ≥ 3. Let ε > 0 and C > 0 be positive numbers. Then there are only finitely many pairs of integers (p, q) with q > 0, which satisfy the inequality*

$$| p/q - b | \leq C/(q^{\tau(d) + \varepsilon})$$

$$\tau(d) = q^{\frac{1}{2}d + 1}$$

Siegel: $\tau(d) = 2\sqrt{d}$

Gelfond, Dyson: $\tau(d) = \sqrt{(2d)}$

Roth: $\tau(d) = 2$

Cubics over finite fields F_p

$$y^2 = x^3 + ax^2 + bx + c$$

is non singular iff $p \neq 2$ and the discriminant

$$D = -4a^3c + a^2b^2 + 18abc - 4b^3 - 27c^2$$

is $\neq 0$.

Let us define

$$C(F_p) = \{ (x,y): x, y \in F_p \text{ and } F(x,y) = 0 \}$$

The number of points of a cubic in F_p

Let us consider a C in F_p: $y^2 = f(x)$

If $f(x) = 0$ then $y=0$.

Among the non zero elements of F_p half of them are squares (the quadratic residues) half not.
Now, imagine to substituite the values of x = 0, 1,…p-1 in $y^2 = f(x)$.

If $f(x) = 0$ there is one solution $y = 0$.

If $f(x) \neq 0$, for half of the possible values non zero of $f(x)$, there are two solutions for y. On the contrary, for the other values there is no solution.

Hence if f(x) were randomly distributed, we would expect of having $p + 1$ solutions.

So $\#C(F_p) = p + 1 +$ errors

Theorem of Hasse (particular case).

For an elliptic curve over F_p,

$$\#C(F_p) = p + 1 + \varepsilon$$

where $| \varepsilon | < 2\sqrt{p}$

Overview

1. A problem of triangles
2. Elliptic curves
3. **Factoring big numbers**
4. Elliptic cryptography

A Factorization Algorithm Using Elliptic Curves (Pollard).

Pollard's method does not work for all n's; but when it does work, it is fairly efficient.

It is the prototype for the elliptic curve method we will discuss later.

The idea underlying Pollard's algorithm:

Suppose that n happens to have a prime factor p such that $p - 1$ is a product of small primes.
From Fermat's little theorem, we know that if p does not divide a, then:

$$a^{p-1} - 1 = 0 \pmod{p}$$

so p will divide gcd $(a^{p-1} - 1, n)$

Of course, at the start we do not know what p is, so we cannot actually compute $a^{(p-1)} - 1$.

Instead, we choose an integer

$k = 2\text{^}e_1 * 3\text{^}e_3 * 5\text{^}e_5 \ldots r\text{^}e_r$

where 2,3,..., r are the first few primes and e_1, e_2..., e_r are small positive integers.

Then we compute (a fast computation):

$$\gcd(a^k - 1, n)$$

Now if we are lucky enough that n has a prime factor p with $(p - 1) \mid k$, then p will divide $a^k - 1$. So in this case we will have

$$\gcd(a^k - 1, n) \geq p > 1.$$

If gcd(a^k - 1, n) $\neq n$, then this gcd provides a non-trivial factor of n; so we can factor n into two pieces and repeat the procedure for each piece.

On the other hand, if the gcd equals n, then we choose a new a and try again.

So the idea is to compute gcd(a^k - 1, n). If it is strictly between 1 and n, we have factored n; if it equals n, we choose a new a; and if it equals 1, we choose a larger k.

Example

Factor $n = 246082373$

The first thing to do is to verify that n is not itself prime. This can be done by computing $2^n - 1 \pmod{n}$ and showing it is not equal to 1.

We will take

$$a = 2$$

and

$$k = 22 \bullet 32 \bullet 5 = 180$$

Since $180 = 2^2 + 2^4 + 2^5 + 2^7$
we need to compute $2\text{^}(2^i) \pmod{n}$ for
$0 < i < 7$.

i	$2^{2^i} \pmod{246082373}$
0	2
1	4
2	16
3	256
4	65536
5	111566955
6	166204404
7	214344997

Using this table, we can compute:

$$\begin{aligned} 2^{180} &= 2^{2^2+2^4+2^5+2^7} \\ &\equiv 16 \cdot 65536 \cdot 111566955 \cdot 28795219 \pmod{246082373} \\ &\equiv 121299227 \pmod{246082373}. \end{aligned}$$

Then a short calculation using the Euclidean algorithm yields:

$$\gcd(2^{180} - 1, n) = \gcd(121299226, 246082373) = 1$$

So the test fails, and n has no prime factors p such that p - 1 divides 180.

But all is not lost, we can just go back and choose a larger k.

For our new k we will take
$k = \text{LCM}[2, 3, \ldots, 9] = 2^3 \bullet 3^2 \bullet 5 \bullet 7 = 2520.$

Since

$$2520 = 2^3 + 2^4 + 2^6 + 2^7 + 2^8 + 2^{11}$$

we need to extend our table a little:

i	$2^{2^i} \pmod{246082373}$
8	111354998
9	82087367
10	7262569
11	104815687

Now we can compute:

$$2^{2520} = 2^{2^3+2^4+2^6+2^7+2^8+2^{11}} \equiv 101220672 \pmod{246082373}$$

Then the Euclidean algorithm (it is fast) yields:

$$\gcd(2^{2520} - 1, n) = \gcd(101220671, 246082373) = 2521$$

So we have found a non-trivial factor of n.

More precisely, we have factored n as

$$n = 246082373 = 2521 \bullet 97613.$$

Pollard's p - 1 Algorithm.

Let $n > 2$ be a composite integer to factor.

1. Choose a number k which is a product of small primes to small powers. For example, take

   k = LCM[l, 2,3,...,K] for some integer K.

2. Choose an arbitrary integer a satisfying $1 < a < n$.

3. Calculate gcd(a,n). If it is strictly greater than 1, then it is a non-trival factor of n. Otherwise:

4. Calculate D = gcd(a^k - 1, n). If $1 < D < n$, then D is a non-trivial factor of n and we. If D = 1, go back to Step 1 and take a larger k. If D = n, go back to Step 2 and choose another a.

Notice that Pollard's algorithm should eventually stop because eventually K in Step 1 will equal ½ (p - 1) for some prime p dividing n, so eventually there will be some p-1 dividing k.

However, if it takes this long to work, then the algorithm will not be practical for large values of n. The algorithm only works in a "reasonable" amount of time if it happens that n has a prime divisor p satisfying p - 1 = product of small primes to small powers.

The running time of this algorithm is
$O(k \times \log B \times \log^2 n)$.

Larger values of k make it run slower, but are more likely to produce a factor.

Assume that $p - 1$, where p is the smallest prime factor of n, can be modelled as a random number of size less than $\sqrt{n}$. One can prove that the probability that the largest factor of such a number is less than $(p-1)^{\varepsilon}$ is roughly $\varepsilon^{-\varepsilon}$; so there is a probability of about $3^{-3} = 1/27$ that a k value of $n^{1/6}$ will yield a factorisation.

Lenstra's algorithm

Pollard's algorithm is based on the fact that the non-zero elements in Z/pZ form a group $(Z/pZ)^*$ of order p - 1; so if $p - 1 \mid k$, then $a^k = 1$ in the group.

Lenstra's algorithm

Lenstra's idea is to replace the group $(Z/pZ)^*$ by the group of points on an elliptic curve $C(F_p)$, and to replace the integer a by a point P of $C(F_p)$.

Arjen K. Lenstra

As in Pollard's algorithm, we choose an integer k composed of a product of small primes.

Then, if it happens that the number of elements in $C(F_p)$ divides k, we will have $k\mathrm{P} = \mathrm{O}$ in $C(F_p)$.

And just as before, the fact that $k\mathrm{P} = \mathrm{O}$ will generally allow us to find a non-trivial factor of n.

Advantages of Lenstra's algorithm

If we choose only one curve C with integer coefficients and consider its reductions modulo various primes, then there is no advantage.

For a single curve C, we will win if there is some prime p dividing n so that $\#C(F_p)$ is a product of small primes.

Similarly, we win using Pollard's algorithm if there is a prime p dividing n so that $p - 1$ is a product of small primes.

But suppose now that we do not win.
Using Pollard's algorithm, not winning means losing, and the game is over.

But with Lenstra's algorithm, there is a new flexibility which allows us to continue playing. Namely, we are free to choose a new elliptic curve and start all over again.

Since $\#C(\mathbb{F}_p)$ varies considerably for a fixed prime p and varying curve C, our odds of eventually winning are fairly good.

Memento. If C is a non-singular cubic curve with coefficients in F_p, then $\#C(F_p) = p + 1 - \varepsilon_p$ with $| \varepsilon_p | \leq 2\sqrt{p}$.

Further, one can show that as C varies over all such curves, the numbers ε_p are quite well spread out over the interval $[-2\sqrt{p}), +2\sqrt{p}]$.

So it is quite likely that we will fairly rapidly run across a curve C with $\#C(F_p)$ equal to a product of small primes.

Lenstra's Elliptic Curve Algorithm.

Let $n \geq 2$ be a composite integer to factor with some special trivial cases checked out.

1. Choose random numbers b, x, y in $]1,n[$
2. Put $c = y^2 - x^3 - bx \pmod{n}$

   C: $y^2 = x^3 + bx + c$;

   and let P (x_1,y_1) belongs to C.

3. Check that $\gcd(4b^3 + 27c^2, n) = 1$. (If it equals n, go back and choose a new b. If it is strictly between 1 and n, then it is a non-trivial factor of n, so we are done.)
4. Choose a number k which is a product of small primes to small powers.

   For example, take k = LCM[1,2,3,...,K], for some K.

Lenstra's Elliptic Curve Algorithm.

5. Compute kP = $(a_k/(d_k)^2, b_k/(d_k)^3)$

6. Calculate D = gcd(d_k,n).

   - If $1 < D < n$, then D is a non-trivial factor of n and we are done.

   - If D = 1, either go back to Step 5 and increase k or go back to Step 2 and choose a new curve.

   - If D = n, then go back to Step 5 and decrease k.

Overview

1. A problem of triangles
2. Elliptic curves
3. Factoring big numbers
4. **Elliptic cryptography**

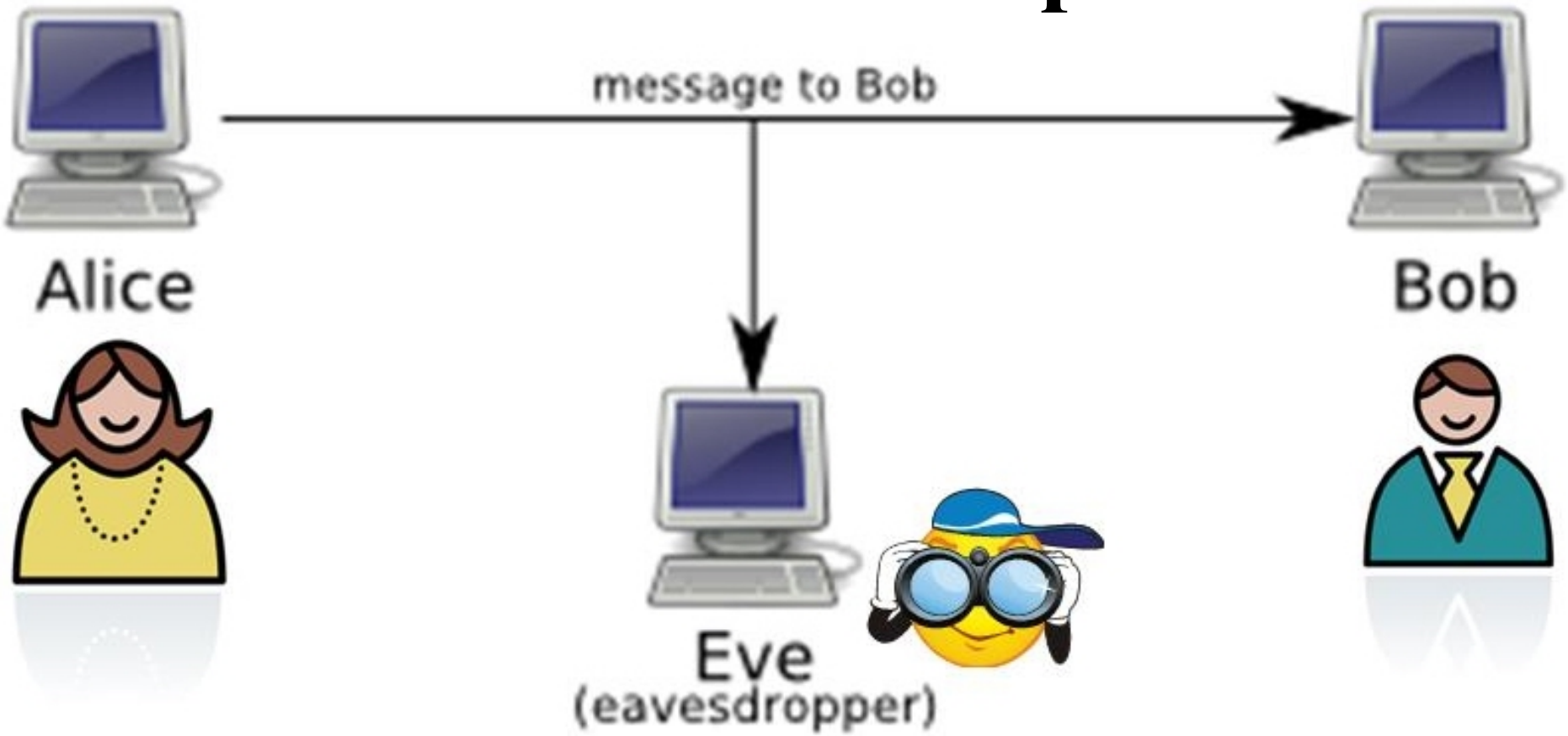
ECC

Neal Koblitz

Victor S. Miller

The use of elliptic curves in cryptography was suggested independently by N K and V S M in 1985. Elliptic curve cryptography algorithms entered wide use in 2004 to 2005.

The Basic Setup



Alice wants to send the **plaintext**, to Bob. In order to keep the eavesdropper Eve from reading the message, she encrypts it to obtain the **ciphertext**. When Bob receives the ciphertext, he decrypts it and reads the message. In order to encrypt the message, Alice uses an **encryption key**. Bob uses a **decryption key** to decrypt the ciphertext. Clearly, the decryption key must be kept secret from Eve.

There are two basic types of encryption:

- **symmetric encryption**

  (Data Encryption Standard – DES – and the Advanced Encryption Standard – AES =*Rijndael* **);**

- **public key encryption**

  (asymmetric encryption).

Generally, public key systems are slower than good symmetric systems. Therefore, it is common to use a public key system to establish a key that is then used in a symmetric system. The improvement in speed is important when massive amounts of data are being transmitted.

Public key encryption are based on the asymmetry between exp and log in F_p.
As it is well known, the former is very fast; the latter (the discrete logarithm) is very slow.

The Discrete Logarithm

Let G be a finite abelian group (written multiplicatively) and let $g \in G$ be a fixed element of known order q.

Let $G_0 = \langle g \rangle$ the cyclic subgroup of G generated by g.

Then we have an isomorphism of groups

$$\exp_g : Z/qZ \to G_0,$$
$$k \to g^k$$

The inverse map of $\exp_g$ is called the *discrete logarithm* (with respect to basis g)
$\log_g : G_0 \rightarrow \mathbb{Z}/q\mathbb{Z}$.

More concretely, given an element $x \in G_0 = \langle g \rangle$, the discrete logarithm of x is the unique number k mod q such that $x = g^k$.

Popular choices for the group G are the multiplicative group of a finite field or the additive group of an elliptic curve over a finite field.

The crucial point for the cryptographical applications is that the exponential map can be effectively calculated, whereas the calculation of the logarithm is in general much more complicated.

To give an idea of the orders of magnitude involved, the bitsize of the number q (which should be a prime) is typically between 160 and 1024 (i.e. $q \approx 2^{160}$ up to $q \approx 2^{1024}$).

The power g^k can be calculated by the repeated squaring algorithm.

If

$$k = \sum_{i=0}^{r} b_i 2^r, \quad b_i \in \{0, 1\}$$

then

$$g^k = \prod_{b_i \neq 0} g^{2^i}$$

and $g^{2\wedge i}$ requires i multiplications.

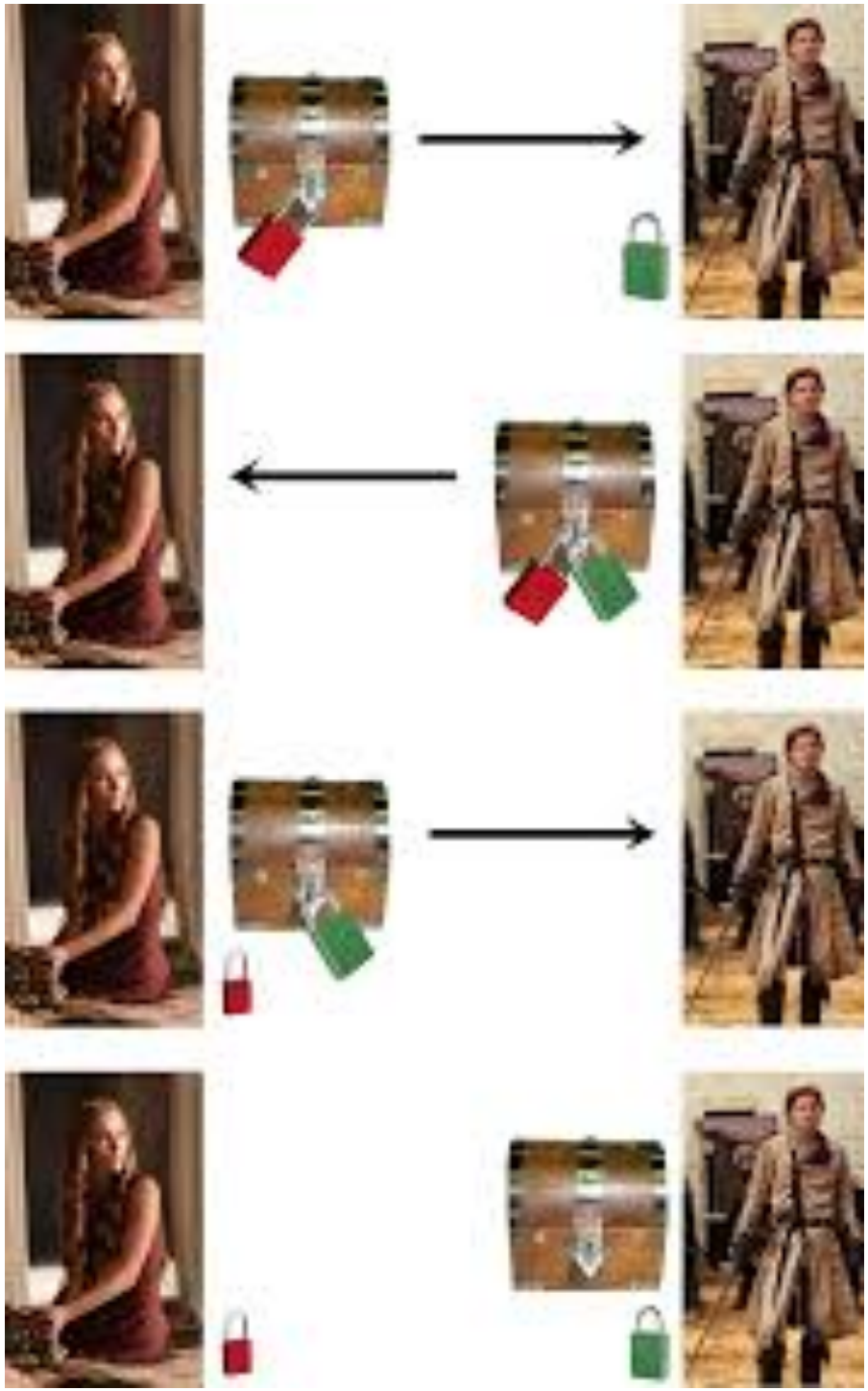
Hence the complexity grows **linearly** with the number of digits of q.

The complexity of the discrete logarithm depends of course on the particular group G.

For general elliptic curves the best known algorithms have a complexity growing **exponentially** with the number of digits of q.

Massey-Omura Encryption

Alice wants to send a message to Bob over public channels. They have not yet established a private key. One way to do this is the following. Alice puts her message in a box and puts her lock on it. She sends the box to Bob. Bob puts his lock on it and sends it back to Alice. Alice then takes her lock off and sends the box back to Bob. Bob then removes his lock, opens the box, and reads the message.



This procedure can be implemented as follows:

1. Alice and Bob agree on an elliptic curve E over a finite field $\mathbf{F}_q$ such that the discrete log problem is hard in $E(\mathbf{F}_q)$. Let $N = \#E(\mathbf{F}_q)$.

2. Alice represents her message as a point $M \in E(\mathbf{F}_q)$.

3. Alice chooses a secret integer m_A with $\gcd(m_A, N) = 1$, computes $M_1 = m_A M$, and sends M_1 to Bob.

4. Bob chooses a secret integer m_B with $\gcd(m_B, N) = 1$, computes $M_2 = m_B M_1$, and sends M_2 to Alice.

5. Alice computes $(m_A)^{-1} \in \mathbf{Z}_N$. She computes $M_3 = (m_A)^{-1} M_2$ and sends M_3 to Bob.

6. Bob computes $(m_B)^{-1} \in \mathbf{Z}_N$. He computes $M_4 = (m_B)^{-1} M_3$. Then $M_4 = M$ is the message.

Formally, we have

$M_4 = (m_B)^{-1}(m_A)^{-1} m_B m_A M = M.$

but we need to justify the fact that $(m_A)^{-1}$, which is an integer representing the inverse of mA mod N, and mA cancel each other.

We have $(m_A)^{-1} m_A \equiv 1 \pmod{N}$.

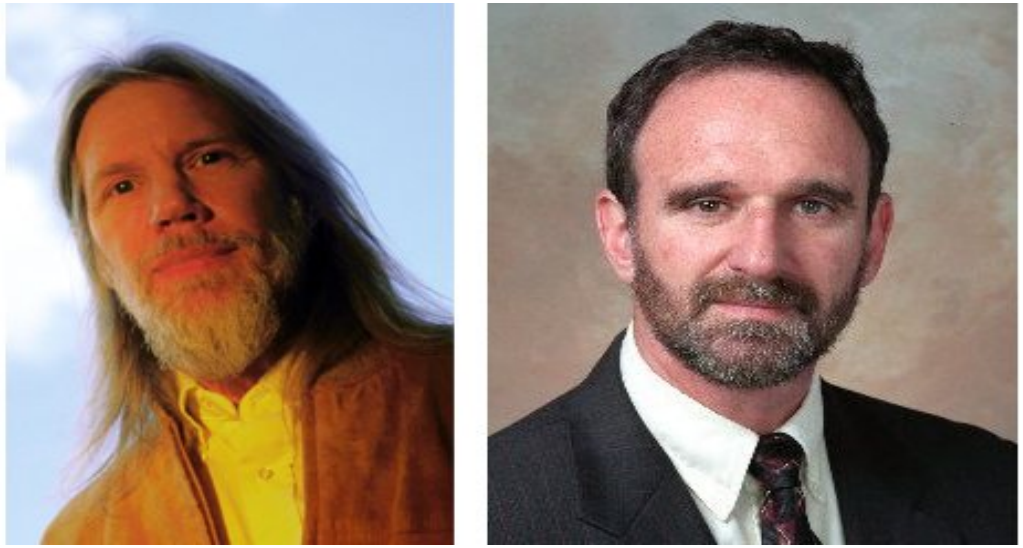
So $(m_A)^{-1} m_A = 1 + kN$ for some k.

The group $E(\mathbf{F}_q)$ has order N, so Lagrange's theorem implies that $NR = \infty$ for any $R \in E(\mathbf{F}_q)$.

Therefore, $(m_A)^{-1} m_A R = (1+kN)R = R + k*\infty = R.$

Applying this to $R = m_B M$, we find that
$M_3 = (m_A)^{-1} m_B m_A \, M = m_B M.$
Similarly, $(m_B)^{-1}$ and m_B cancel , so
$M_4 = (m_B)^{-1} \, M_3 = (m_B)^{-1} m_B \, M = M.$

The eavesdropper Eve knows $E(\mathbf{F}_q)$ and the points $m_A M$, $m_B m_A M$, and $m_B M$.
Let $a = (m_A)^{-1}$, $b = (m_B)^{-1}$, $P = m_A m_B M$. Then we see that Eve knows P, bP, aP and wants to find abP.
This is the Diffie-Hellman problem, which is hard.
The above procedure works in any finite group.



Whitfield Diffie and Martin Hellman

It remains to show how to represent a message as a point on an elliptic curve.

We use a method proposed by Koblitz. Suppose E is an elliptic curve given by

$$y^2 = x^3 + Ax + B \text{ over } \mathbf{F}_p.$$

Let m be a message, expressed as a number $0 \leq m < p/100$. Let $x_j = 100m + j$ for $0 \leq j < 100$.

For $j = 0, 1, 2, \ldots, 99$, compute $s_j = (x_j)^3 + Ax_j + B$.

If $(s_j)^{(p-1)/2} \equiv 1 \pmod{p}$,
then s_j is a square mod p, in which case we do not need to try any more values of j.

When $p \equiv 3 \pmod{4}$, a square root of s_j is then given by $y_j \equiv (s_j)^{(p+1)/4} \pmod{p}$.

When $p \equiv 1 \pmod{4}$, a square root of s_j can also be computed, but the procedure is more complicated.

We obtain a point (x_j, y_j) on E. To recover m from (x_j, y_j), simply compute $[xj/100]$

Since s_j is essentially a random element of $(\mathbf{F}_p)^{\times}$, which is cyclic of even order, the probability is approximately 1/2 that s_j is a square.
So the probability of not being able to find a point for m after trying 100 values is around 2^{-100}.

One might wonder why elliptic curves are used in cryptographic situations.

The main reason is that elliptic curves provide security equivalent to classical systems while using **fewer bits**.

Lenstra recently introduced the concept of **Global Security**. You can compute how much energy is needed to break a cryptographic algorithm, and compare that with how much water that energy could boil. This is a kind of cryptographic carbon footprint. By this measure, breaking a 228-bit RSA key requires less energy to than it takes to boil a teaspoon of water. Comparatively, breaking a 228-bit elliptic curve key requires enough energy to boil all the water on earth. For this level of security with RSA, you'd need a key with 2,380-bits.

Hence with ECC one can use smaller keys to get the same levels of security.
Small keys are important, especially in a world where more and more cryptography is done on less powerful devices like mobile phones.
While you could likely continue to keep RSA secure by increasing the key length that comes with a cost of slower cryptographic performance on the client. ECC appears to offer a better tradeoff: high security with short, fast keys.

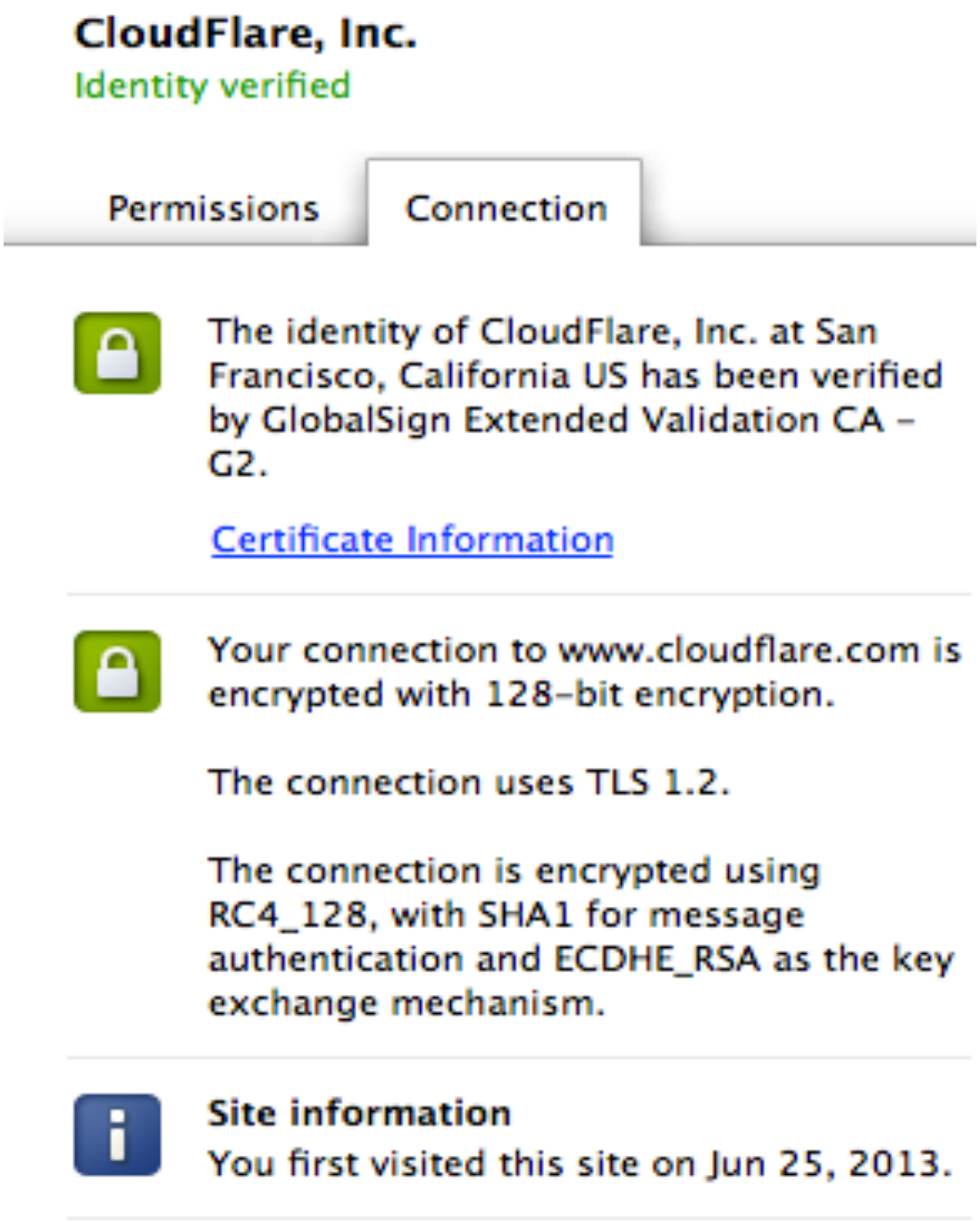
Elliptic curve cryptography is now used in a wide variety of applications: the U.S. government uses it to protect internal communications, it is the mechanism used to prove ownership of bitcoins, it provides signatures in Apple's iMessage service, it is used to encrypt DNS information with DNSCurve, and it is the preferred method for authentication for secure web browsing over SSL/TLS.

CloudFlare uses elliptic curve cryptography to provide perfect forward secrecy which is essential for online privacy.

First generation cryptographic algorithms like RSA and Diffie-Hellman are still the norm in most arenas, but elliptic curve cryptography is quickly becoming the go-to solution for privacy and security online.

If you access the HTTPS version of the blog https://blog.cloudflare.com from a recent enough version of Chrome or Firefox, your browser is using elliptic curve cryptography. You can check this yourself.
In Chrome, you can click on the lock in the address bar and go to the connection tab to see which cryptographic algorithms were used in establishing the secure connection.

Clicking on the lock in the Chrome should show the following image:



The relevant portions of this text to this discussion is ECDHE_RSA. ECDHE stands for Elliptic Curve Diffie Hellman Ephemeral and is a key exchange mechanism based on elliptic curves.
This algorithm is used by CloudFlare to provide perfect forward secrecy in SSL. The RSA component means that RSA is used to prove the identity of the server.

CloudFlare's ECC curve for ECDHE (This is the same curve used by Google.com):

curve: $y^2 = x^3 + ax + b$

a =
115792089210356248762697446949407573530
086143415290314195533631308867097853948

b =
410583637251521421293261297800472684091
144410159937255548352563140394674012 91

The moral of our story

We have encountered a fundamental idea in modern mathematics: the idea of solving a problem about a particular type of object by situating the object in a more general space and finding the right way of parameterizing that space.

This modeling action has allowed us to enter into a rich field of mathematical theories and problems, which currently have also an important follow-up in applications.

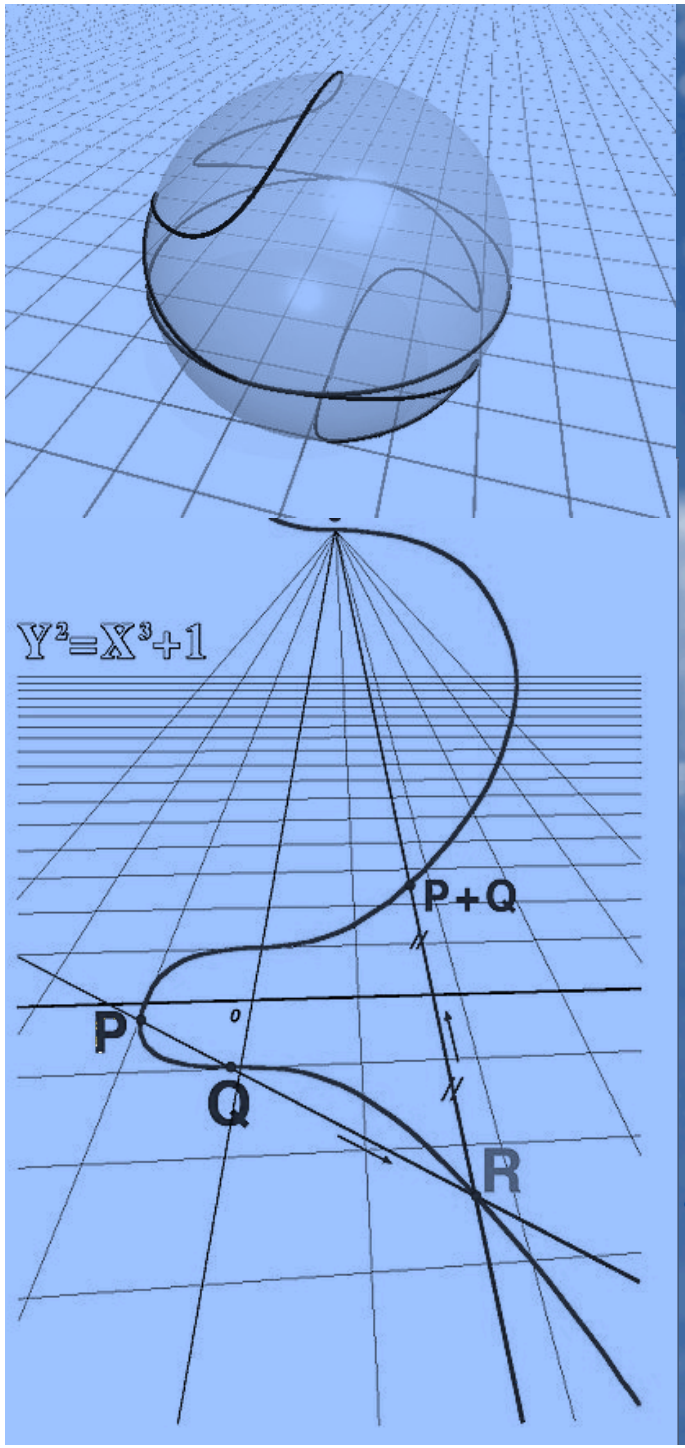
References

Blake, I., Seroussi, G. and Smart, N., editors, (2005). *Advances in Elliptic Curve Cryptography*, London Mathematical Society 317, Cambridge University Press.

Husemöller, D. (2004). *Elliptic Curves*. Second Edition. Springer: New York.

Silverman, J.H., and Tate J. (1992). *Rational Points on Elliptic Curves*. Springer: New York.

Washington, L. (2003). *Elliptic Curves: Number Theory and Cryptography*, Chapman & Hall / CRC.



Dank voor uw aandacht!

Grazie per l'attenzione!