isogenies

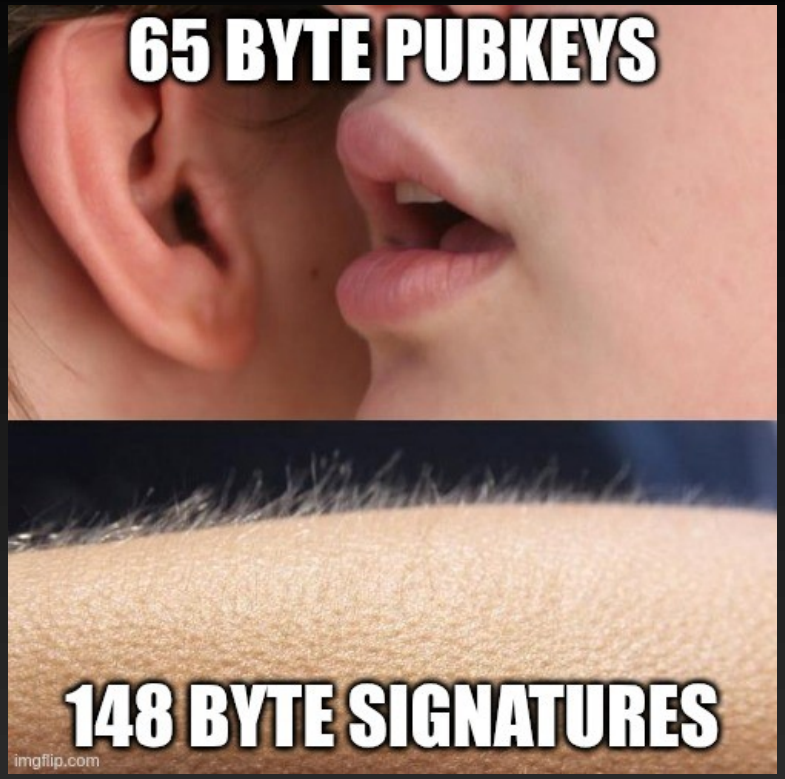
in 30 memes or less

conduition :: https://conduition.io

made possible by a Brink grant

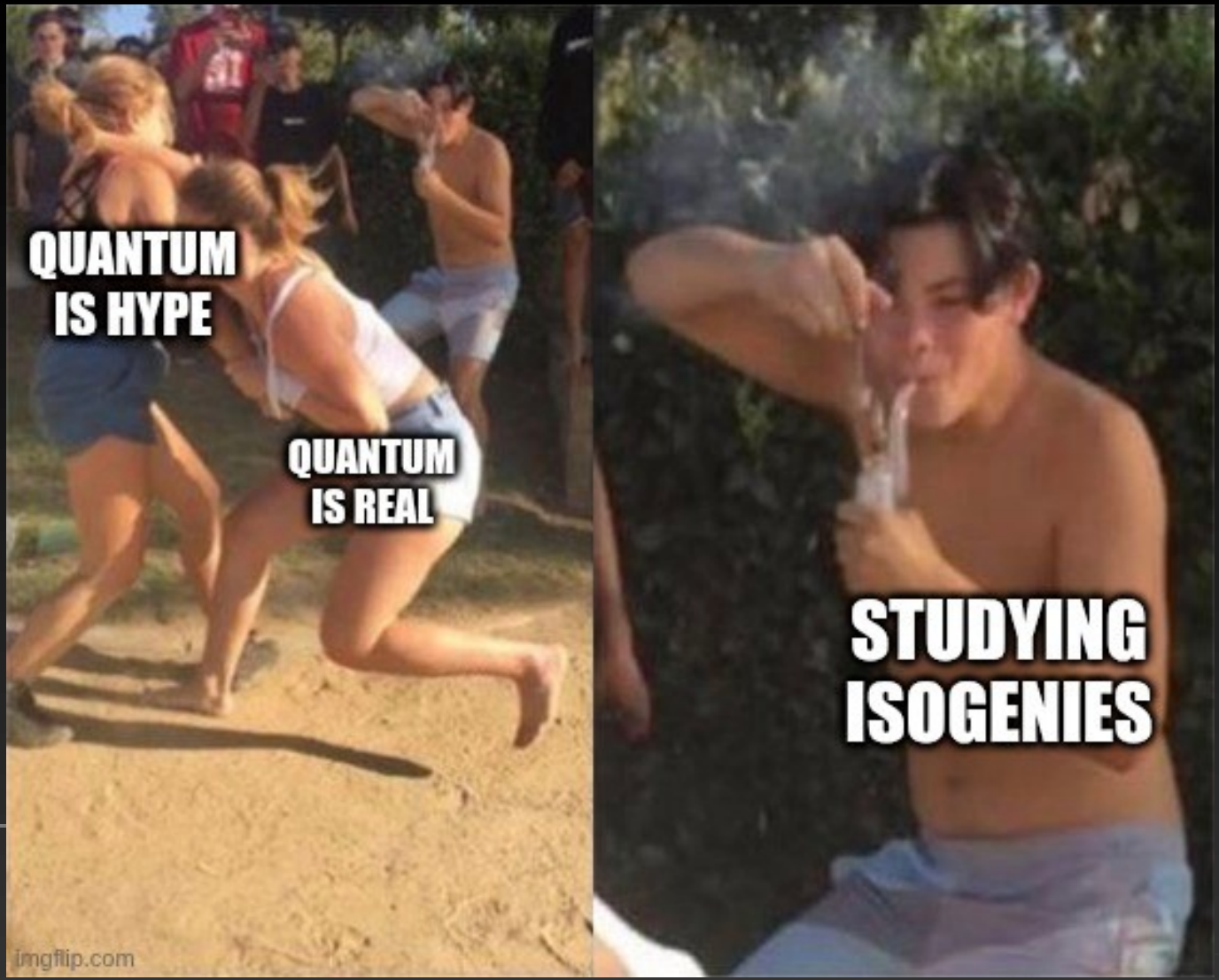
https://conduition.io

why isogenies?

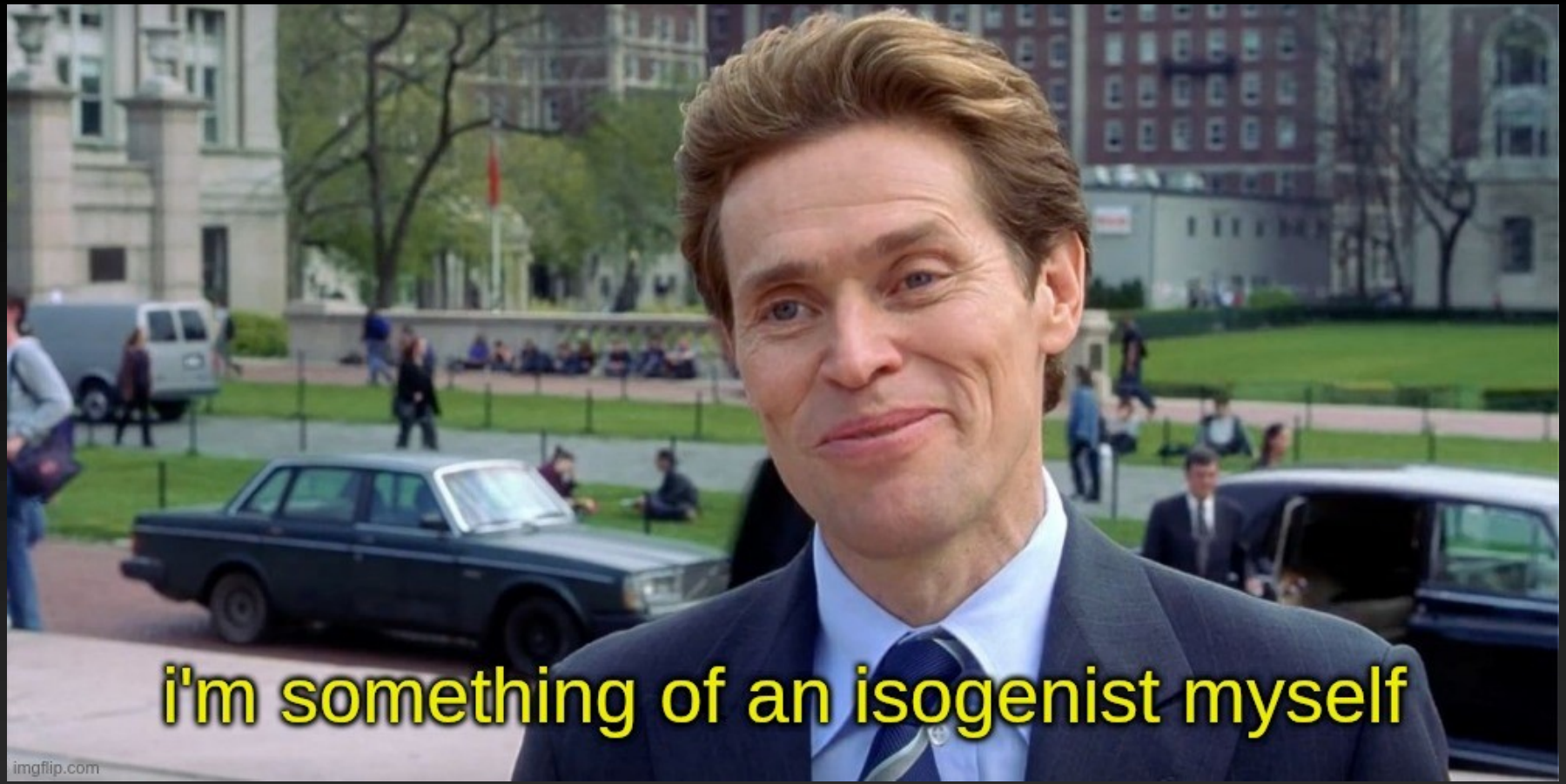


https://conduition.io

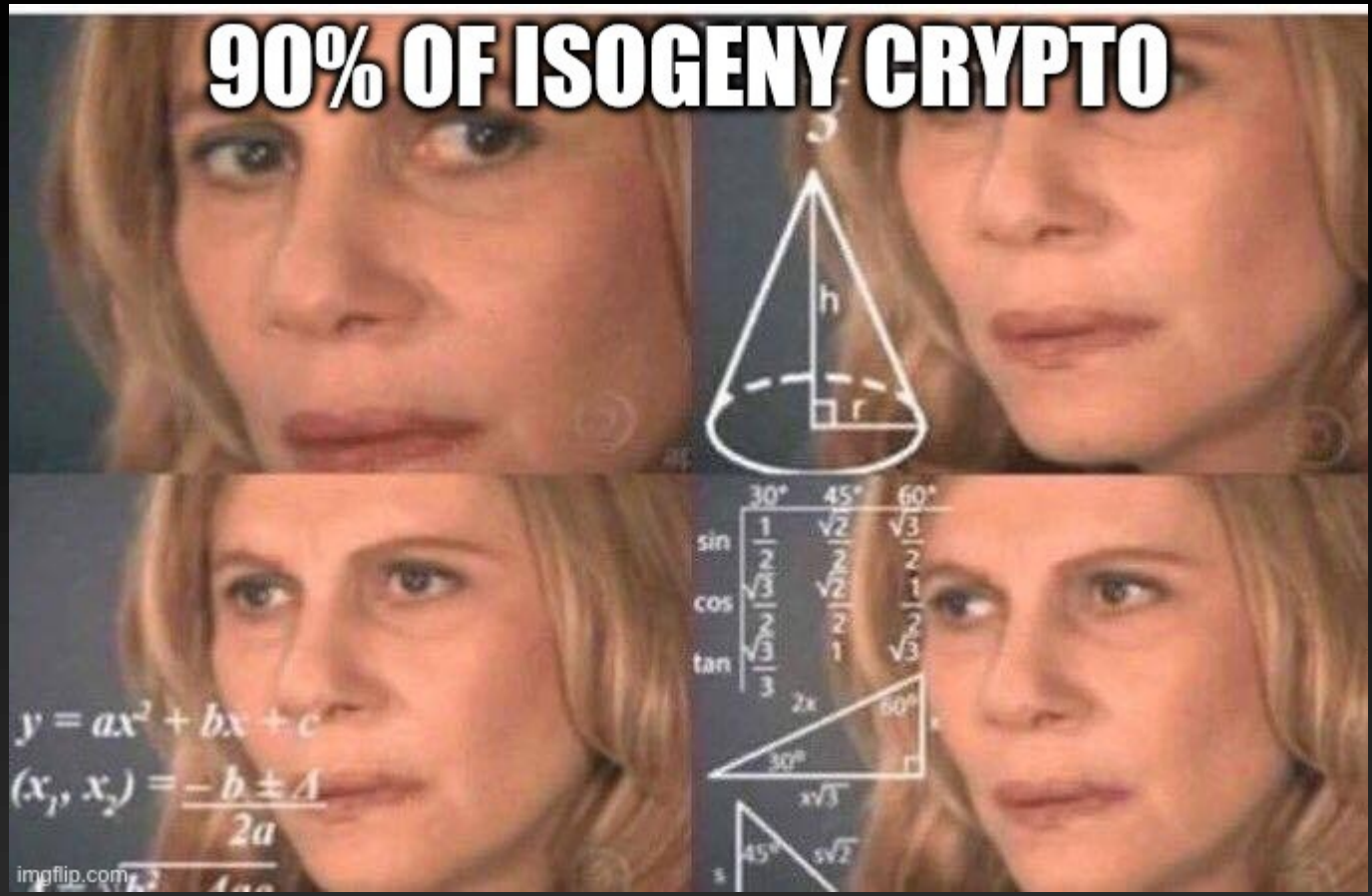
current situation



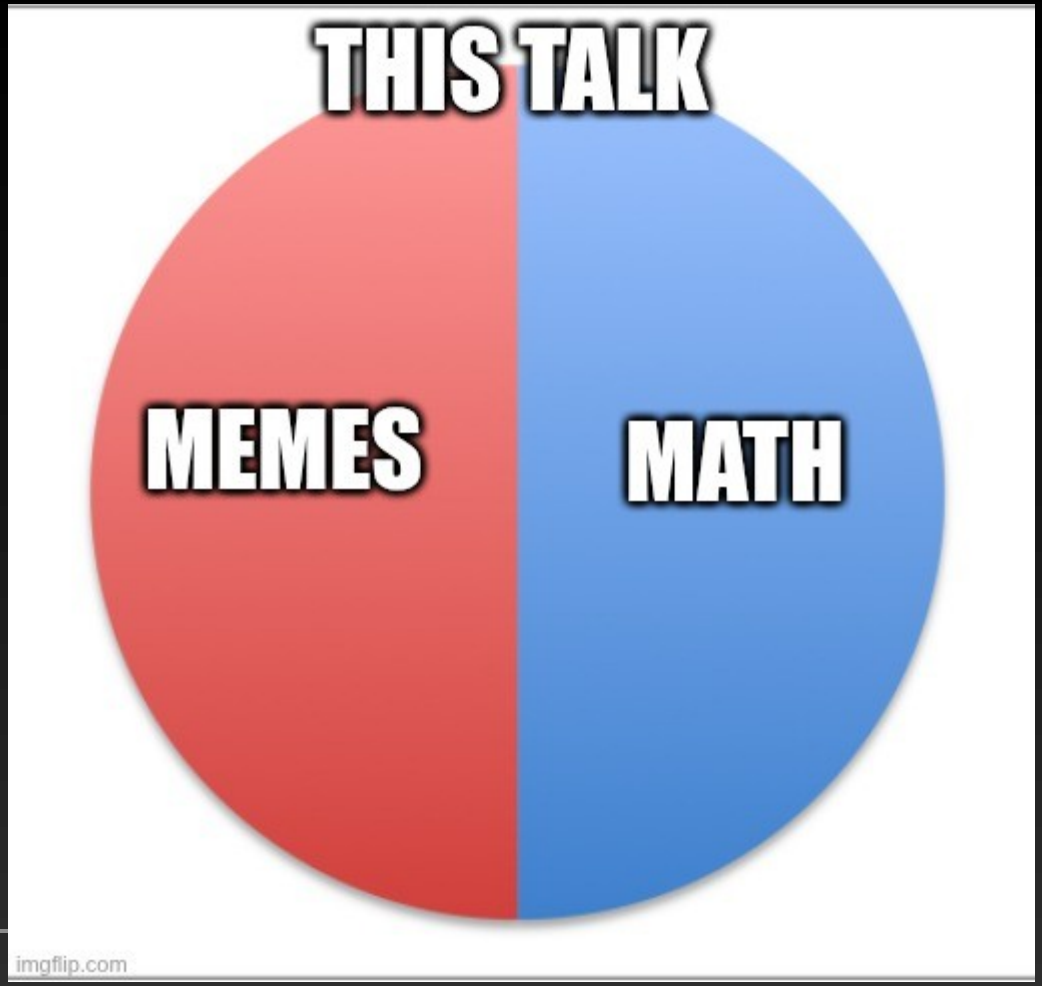
https://conduition.io



https://conduition.io



https://conduition.io



https://conduition.io

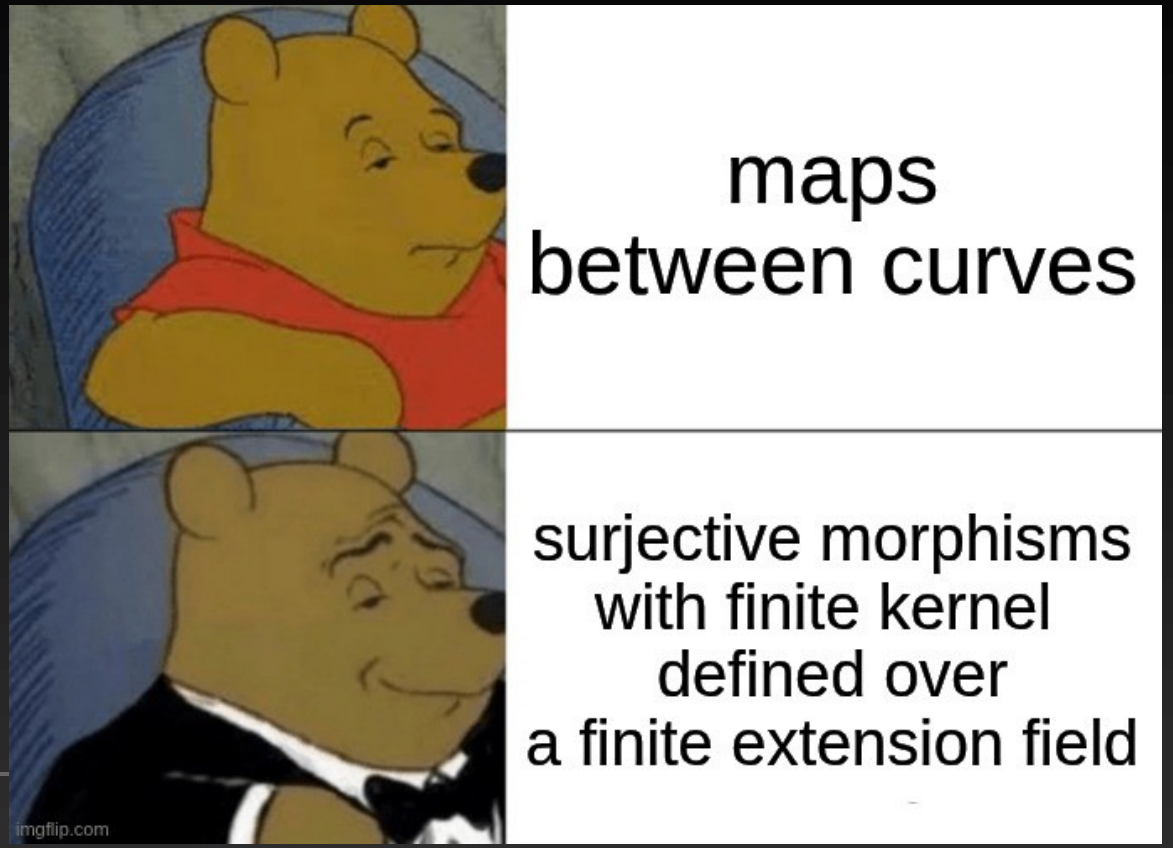
what you'll actually be learning

https://conduition.io

isogenies

how you should think about them

formal definition



https://conduition.io

isogenies

(the math part)

$$\varphi : E_1 \to E_2$$

isogeny
(function)

domain
(input)

codomain
(output)

https://conduition.io

isogenies

evaluation

$$\varphi(P) \in E_2$$

isogeny
(function)

input point

codomain
(output)

https://conduition.io

isogeny composition

$$\varphi_1 : E_1 \to E_2$$

matching (co)domain

$$\varphi_2 : E_2 \to E_3$$

$$E_1 \xrightarrow{\varphi_1} E_2 \xrightarrow{\varphi_2} E_3$$

$$\varphi_2 \circ \varphi_1$$

composition operator
i.e:

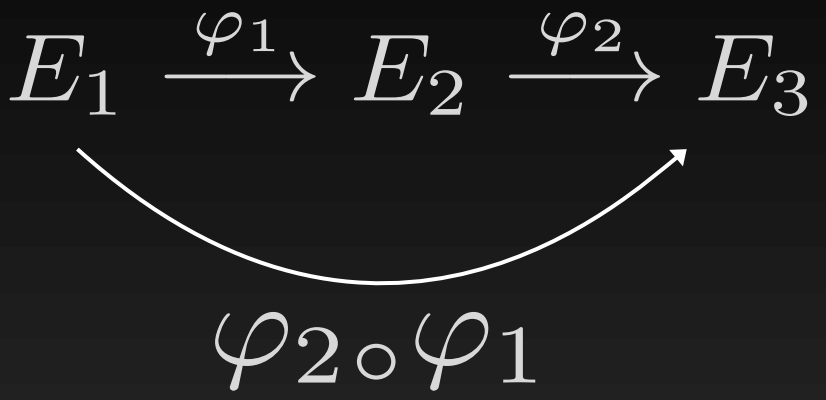
$$(\varphi_2 \circ \varphi_1)(P) = \varphi_2(\varphi_1(P))$$

https://conduition.io

isogeny degrees

a measure of complexity

$$\deg(\varphi_2 \circ \varphi_1) = \deg(\varphi_2) \cdot \deg(\varphi_1)$$



degrees multiply when isogenies are composed (just like polynomials!)

duals

kind of an inverse-ish

$$\varphi : E_1 \to E_2$$

dual isogeny ⟶ $\hat{\varphi} : E_2 \to E_1$

easy to find, given φ

duals are not inverses because:

$$\hat{\varphi}(\varphi(P)) = \deg(\varphi) \cdot P$$

https://conduition.io

isomorphism

curve equivalency

isogeny of degree 1 ⟶ “isomorphism”

if there exists $\tau : E_1 \to E_2$

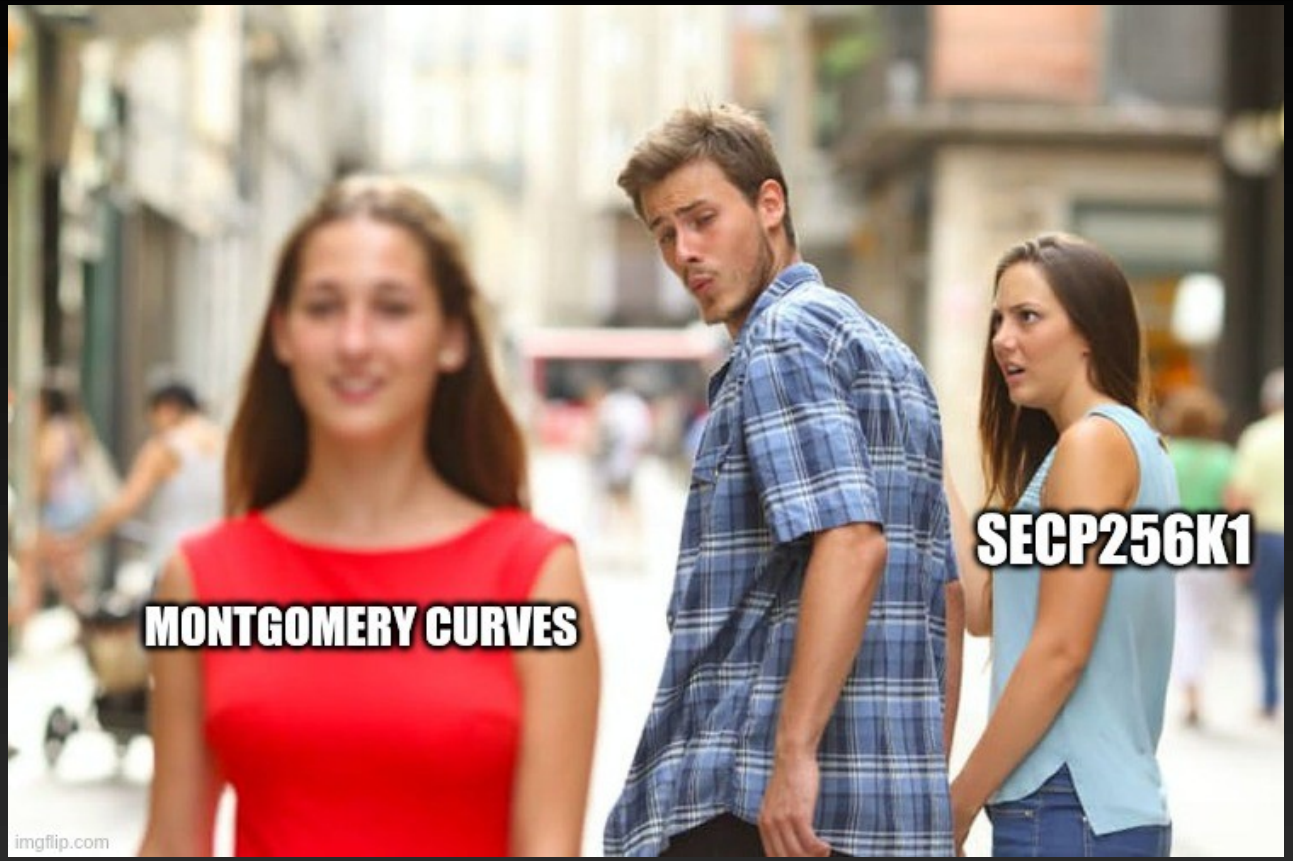
such that $\deg(\tau) = 1$ (an isomorphism)

then $E_1 \cong E_2$ (the curves are *isomorphic* to one another)

easy to find an isomorphism if you know E_1 and E_2

https://conduition.io

isogenies get around



https://conduition.io

isogeny computation

https://conduition.io

supersingular curves

https://en.wikipedia.org/wiki/Supersingular_elliptic_curve

$$|E(\mathbb{F}_{p^k})| \equiv 1 \mod p$$

prime

secp256k1 is not supersingular BTW

https://conduition.io

Definition [edit]

There are many different but equivalent ways of defining supersingular elliptic curves that have been used. Some of the ways of defining them are given below. Let K be a field with algebraic closure $\overline{K}$ and E an elliptic curve over K.

- The $\overline{K}$-valued points $E(\overline{K})$ have the structure of an abelian group. For every n, we have a multiplication map $[n] : E \to E$. Its kernel is denoted by $E[n]$. Now assume that the characteristic of K is $p > 0$. Then one can show that either

  $$E[p^r](\overline{K}) \cong \begin{cases} 0, & \text{or} \\ \mathbb{Z}/p^r\mathbb{Z} \end{cases}$$

  for $r = 1, 2, 3, \ldots$ In the first case, E is called *supersingular*. Otherwise it is called *ordinary*. In other words, an elliptic curve is supersingular if and only if the group of geometric points of order p is trivial.
- Supersingular elliptic curves have many endomorphisms over the algebraic closure $\overline{K}$ in the sense that an elliptic curve is supersingular if and only if its endomorphism algebra (over $\overline{K}$) is an order in a quaternion algebra. Thus, their endomorphism algebra (over $\overline{K}$) has rank 4, while the endomorphism group of every other elliptic curve has only rank 1 or 2. The endomorphism ring of a supersingular elliptic curve can have rank less than 4, and it may be necessary to take a finite extension of the base field K to make the rank of the endomorphism ring 4. In particular the endomorphism ring of an elliptic curve over a field of prime order is never of rank 4, even if the elliptic curve is supersingular.
- Let G be the formal group associated to E. Since K is of positive characteristic, we can define its height ht(G), which is 2 if and only if E is supersingular and else is 1.
- We have a Frobenius morphism $F : E \to E$, which induces a map in cohomology
  $F^* : H^1(E, \mathcal{O}_E) \to H^1(E, \mathcal{O}_E)$.

  The elliptic curve E is supersingular if and only if F^* equals 0.
- We have a Verschiebung operator $V : E \to E$, which induces a map on the global 1-forms
  $V^* : H^0(E, \Omega^1_E) \to H^0(E, \Omega^1_E)$.

  The elliptic curve E is supersingular if and only if V^* equals 0.
- An elliptic curve is supersingular if and only if its Hasse invariant is 0.
- An elliptic curve is supersingular if and only if the group scheme of points of order p is connected.
- An elliptic curve is supersingular if and only if the dual of the Frobenius map is purely inseparable.
- An elliptic curve is supersingular if and only if the "multiplication by p" map is purely inseparable and the j-invariant of the curve lies in a quadratic extension of the prime field of K, a finite field of order p^2.
- Suppose E is in Legendre form, defined by the equation $y^2 = x(x-1)(x-\lambda)$, and p is odd. Then for $\lambda \neq 0, 1$, E is supersingular if and only if the sum

  $$\sum_{i=0}^{n} \binom{n}{i}^2 \lambda^i$$

  vanishes, where $n = (p-1)/2$. Using this formula, one can show that there are only finitely many supersingular elliptic curves over K (up to isomorphism).
- Suppose E is given as a cubic curve in the projective plane given by a homogeneous cubic polynomial $f(x,y,z)$. Then E is supersingular if and only if the coefficient of $(xyz)^{p-1}$ in f^{p-1} is zero.
- If the field K is a finite field of order q, then an elliptic curve over K is supersingular if and only if the trace of the q-power Frobenius endomorphism is congruent to zero modulo p.

  When $q=p$ is a prime greater than 3 this is equivalent to having the trace of Frobenius equal to zero (by the Hasse bound); this does not hold for p=2 or 3.

why supersingular curves?

(don’t worry if you don’t understand this yet)

1. Every S.S curve is isomorphic to a S.S curve defined over $\mathbb{F}_{p^2}$

2. Every S.S curve defined over $\mathbb{F}_{p^2}$ is connected to every other such curve by an isogeny defined over $\mathbb{F}_{p^2}$

3. Any isogeny can be factored into a *chain of isogenies* with prime degrees.

https://conduition.io

supersingular isogeny graph

vertices are isomorphism classes (buckets of like curves)

edges are prime-degree isogenies (maps between curves)

For a large prime p there are about $\frac{p}{12}$ vertices (isomorphism classes)

image credit: Craig Costello, 2019

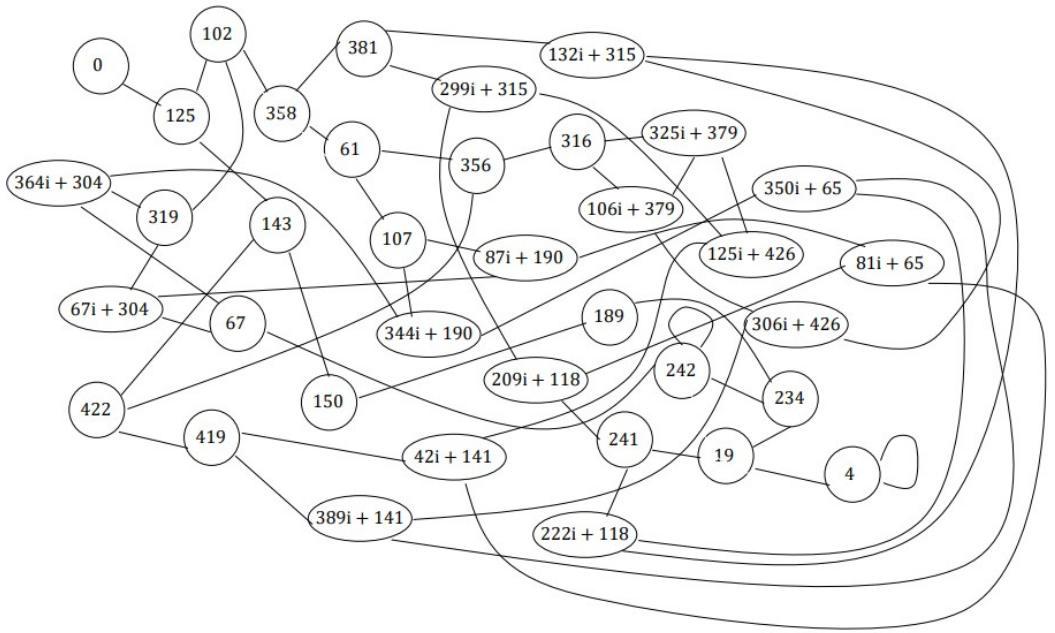


Fig. 4. The 2-isogeny graph for $p = 431$. The 37 nodes are the supersingular j-invariants and the edges between them correspond to 2-isogenies.

https://conduition.io

supersingular isogeny path problem (SIPP)

the isogenist's BFF

given two supersingular elliptic curves E_1, E_2 s. t $E_1 \ncong E_2$

find an isogeny $\varphi : E_1 \to E_2$

many candidates exist,
but hard to find any of them

best general algorithm:

Meet-in-the-middle

complexity:

classical: $O(\sqrt{p})$

quantum: $O(p^{\frac{1}{4}})$

https://conduition.io

endomorphisms

map a point on a curve to another point on the same curve

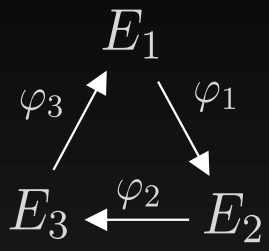
examples:

$$m \in \mathbb{Z}$$
$$mP : P \in E_1$$
$$[m] : E_1 \to E_1$$

point multiplication

$$q = p^k$$
$$\pi((x, y)) = (x^q, y^q)$$
$$\pi : E_1 \to E_1$$

frobenius map



$$\varphi = \varphi_3 \circ \varphi_2 \circ \varphi_1$$
$$\varphi : E_1 \to E_1$$

composed isogenies

there's a **finite set of endomorphisms** available on any given curve,

https://conduition.io

and they form a ring: an **endomorphism ring!**

endomorphism ring problem (ERP)

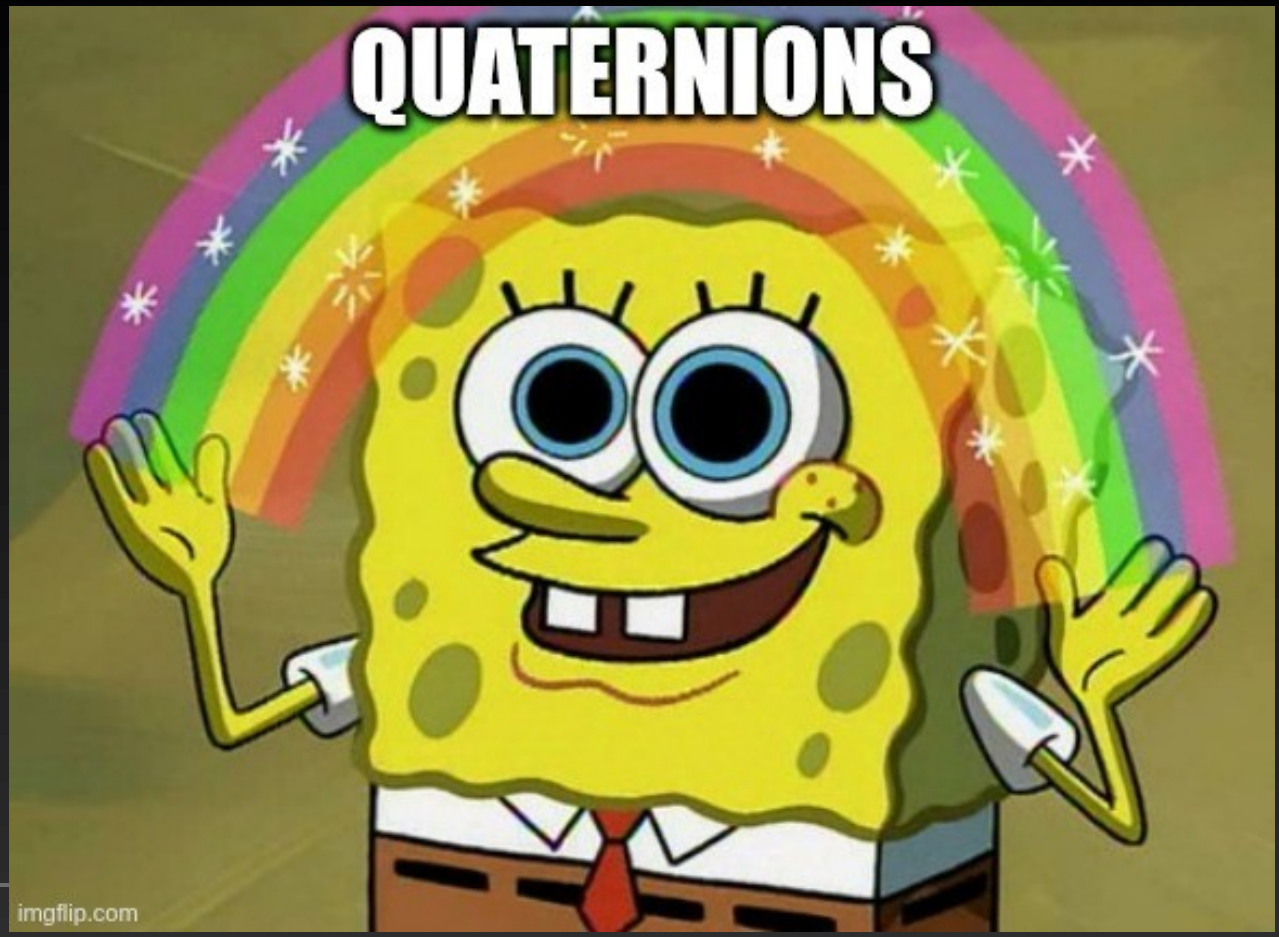
given an elliptic curve E

find a generator of its endomorphism ring, End(E)

apparently hard even for a QC

https://conduition.io

endomorphism rings are closely related to **quaternion algebras**



https://conduition.io

ERP and SIPP are equivalent

https://conduition.io

abusing SIPP/ERP equivalence

two S.S. elliptic curves E_0, E_1

assume $\mathrm{End}(E_0)$ is known

if $\mathrm{End}(E_1)$ is known

then we can find an isogeny

$$\varphi : E_0 \to E_1$$

if an isogeny $\varphi : E_0 \to E_1$ is known

we can find the endomorphism ring

$$\mathrm{End}(E_1)$$

computable with KLPT!

https://conduition.io

toy example

key generation

base S.S. elliptic curve E_0

assume $\mathrm{End}(E_0)$ is publicly known

generate a random isogeny $\varphi : E_0 \to E_{\mathrm{pk}}$

compute the endomorphism ring $\mathrm{End}(E_{\mathrm{pk}})$

E_{pk} **is our public key** ⟶

very compact if we use Montgomery curves!

$$y^2 = x^3 + Ax^2 + x$$

(we only need to publish the coefficient A)

$\mathrm{End}(E_{\mathrm{pk}})$ **(or** φ **) is our secret key**

https://conduition.io

toy example

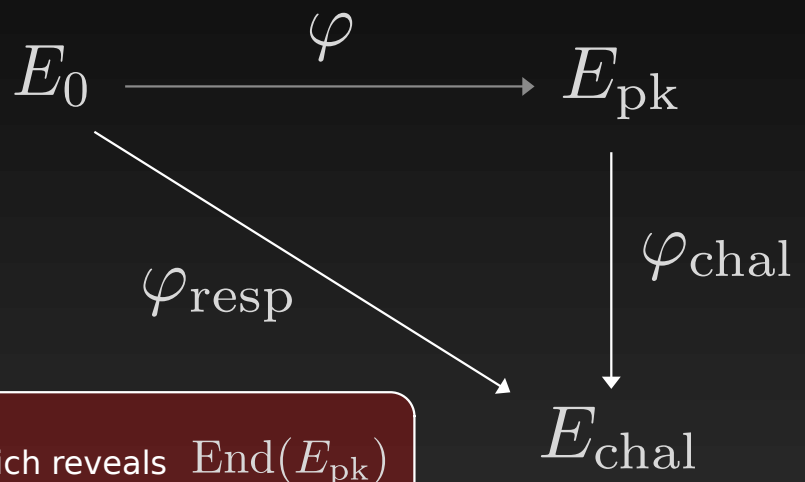
signing

base S.S. elliptic curve E_0

public key curve E_{pk}

secret isogeny $\varphi : E_0 \to E_{\mathrm{pk}}$

1. derive challenge isogeny $\varphi_{\mathrm{chal}} = H(E_{\mathrm{pk}}, m)$
2. compute $\mathrm{End}(E_{\mathrm{chal}})$ (using $\mathrm{End}(E_{\mathrm{pk}}), \varphi_{\mathrm{chal}}$)
3. find φ_{resp} (using $\mathrm{End}(E_0), \mathrm{End}(E_{\mathrm{chal}})$)



φ_{resp} **is our signature**

problem: leaks an isogeny $\hat{\varphi}_{\mathrm{chal}} \circ \varphi_{\mathrm{resp}} : E_0 \to E_{\mathrm{pk}}$ which reveals $\mathrm{End}(E_{\mathrm{pk}})$

toy example

signing

base S.S. elliptic curve E_0

public key curve E_{pk}

secret isogeny $\varphi : E_0 \to E_{\mathrm{pk}}$

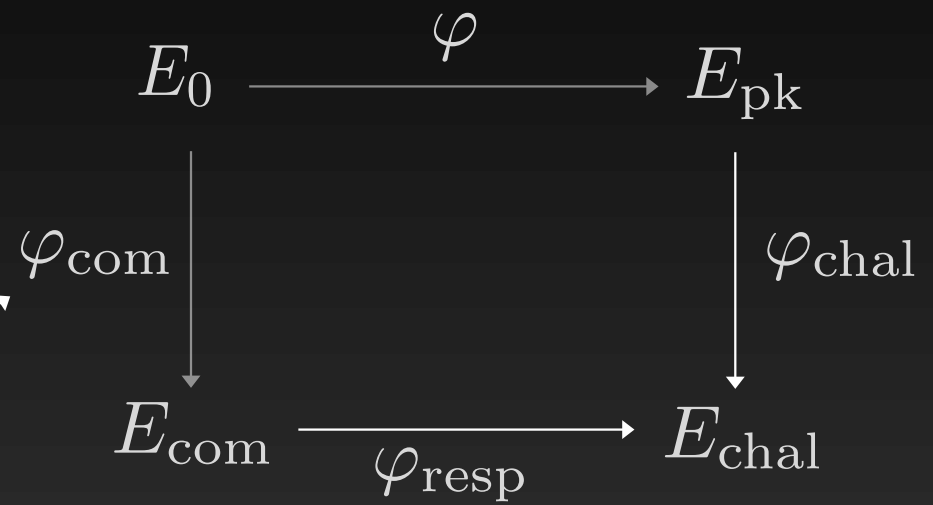
0. sample random commitment isogeny φ_{com} & compute $\mathrm{End}(E_{\mathrm{com}})$

1. derive challenge isogeny $\varphi_{\mathrm{chal}} = H(E_{\mathrm{com}}, E_{\mathrm{pk}}, m)$

2. compute $\mathrm{End}(E_{\mathrm{chal}})$ (using $\mathrm{End}(E_{\mathrm{pk}}), \varphi_{\mathrm{chal}}$)

3. find φ_{resp} (using $\mathrm{End}(E_{\mathrm{com}}), \mathrm{End}(E_{\mathrm{chal}})$)

$E_{\mathrm{com}}, \varphi_{\mathrm{resp}}$ **is our signature**



now zero knowledge because φ_{com} is kept secret

https://conduition.io

protip: reusing φ_{com} will leak $\mathrm{End}(E_{\mathrm{pk}})$

https://conduition.io

SQIsign ~~toy example~~

signing

base S.S. elliptic curve E_0

public key curve E_{pk}

secret isogeny $\varphi : E_0 \to E_{\mathrm{pk}}$

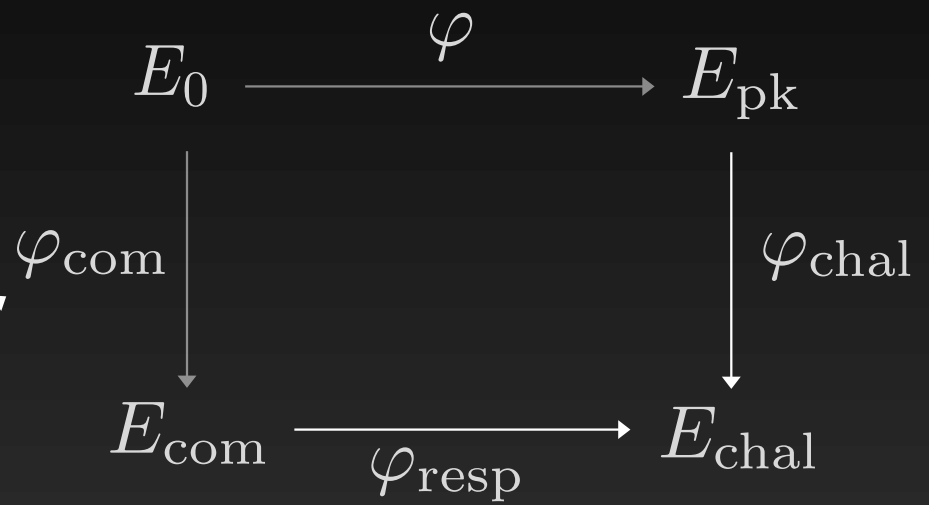
0. sample random commitment isogeny φ_{com} & compute $\mathrm{End}(E_{\mathrm{com}})$

1. derive challenge isogeny $\varphi_{\mathrm{chal}} = H(E_{\mathrm{com}}, E_{\mathrm{pk}}, m)$

2. compute $\mathrm{End}(E_{\mathrm{chal}})$ (using $\mathrm{End}(E_{\mathrm{pk}}), \varphi_{\mathrm{chal}}$)

3. find φ_{resp} (using $\mathrm{End}(E_{\mathrm{com}}), \mathrm{End}(E_{\mathrm{chal}})$)

$E_{\mathrm{com}}, \varphi_{\mathrm{resp}}$ **is our signature**



now zero knowledge because φ_{com} is kept secret

https://conduition.io

protip: reusing φ_{com} will leak $\mathrm{End}(E_{\mathrm{pk}})$

isogeny encoding

why signatures are so small

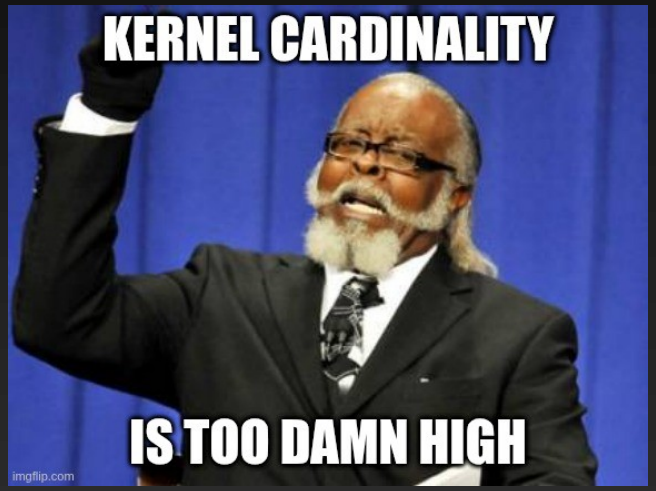
any *separable** isogeny: $\varphi : E_1 \to E_2$

can be recovered from: $\ker(\varphi) = \{P \in E_1 : \varphi(P) = 0_{E_2}\}$

$= \{0_{E_1}, P_2, P_3, \ldots P_{\deg(\varphi)}\}$

always includes infinity point

a finite set of points!



bad news: we typically use isogenies w/ very big kernels

$$\deg(\varphi) = |\ker(\varphi)| \geq 2^{128}$$

kernel is too big to be our encoding

https://conduition.io

*any *inseparable* isogeny can be made *separable*

isogeny encoding

why signatures are so small

any *separable** isogeny: $\varphi : E_1 \to E_2$

can be recovered from: $\ker(\varphi) = \{P \in E_1 : \varphi(P) = 0_{E_2}\}$

$= \{0_{E_1}, P_2, P_3, \ldots P_{\deg(\varphi)}\}$

↑ always includes infinity point

↑ a finite set of points!

good news: we can use torsion groups to represent very big sets of points

$$\ker(\varphi) \leftarrow E_1[\deg(\varphi)]$$

https://conduition.io

*any *inseparable* isogeny can be made *separable*

isogeny encoding

why signatures are so small

t-torsion group: $E[t] = \{P \in E : tP = 0_E\}$ size: $|E[t]| = t^2$

$$P \in E[t] \longrightarrow \overbrace{P + P + P + \ \ldots \ + P}^{t \text{ times}} = 0_E$$

every point in the t-torsion group is a *generator* of a *cyclic subgroup* of order n | t :

"subgroup generated by P" $\longrightarrow \langle P \rangle = \{0_E,\ P,\ 2P,\ 3P,\ \ldots\ (n-1)P\}$ (all points unique)

one point efficiently encodes a large set of points!
sounds like exactly what we need

important distinction:

"P is a t-torsion point" → P has order n that **divides** t
"P has order n" → P generates n unique points

https://conduition.io


YOU SAID THERE
WOULD BE MORE MEMES
imgflip.com
I SAID 30
MEMES OR LESS

isogeny encoding

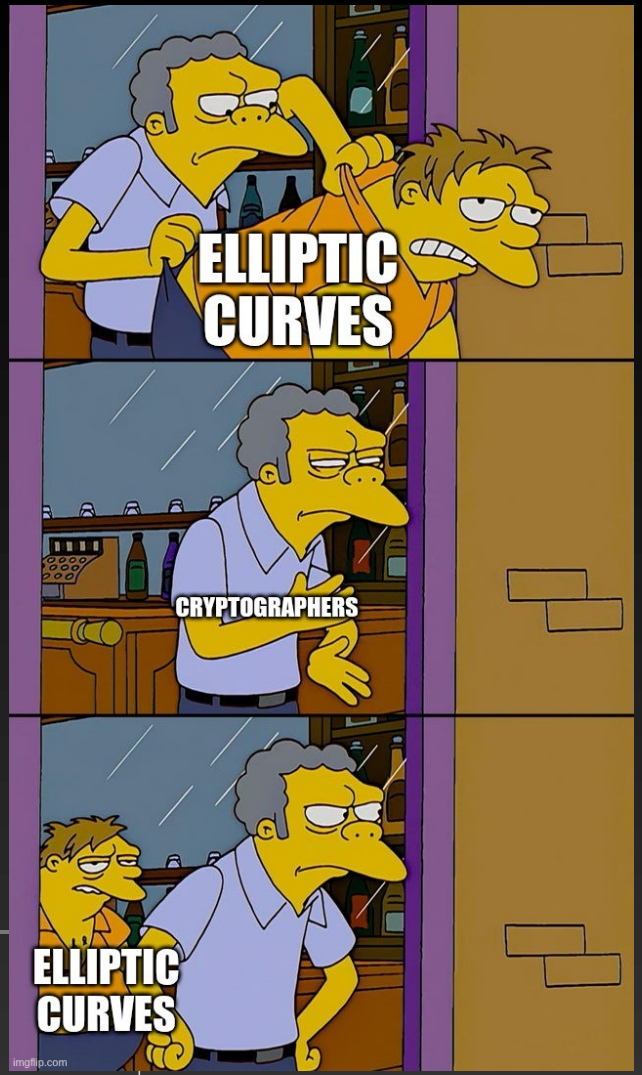
why signatures are so small

- ϕ: $E_1 \to E_2$ can be encoded as $\ker(\phi)$
- if $\ker(\varphi) = \langle P \rangle$, then $\ker(\phi)$ can be encoded with just P.
- otherwise (ϕ is not “cyclic”) we can decompose ϕ into a distinct cyclic isogeny ψ: $E_1 \to E_2$ such that $\ker(\psi) = \langle P \rangle$ and we can encode $\ker(\psi)$ with just P.

upshot: **we can encode *an* isogeny between any two curves with a single point**

not necessarily *any* isogeny

https://conduition.io



https://conduition.io

rerandomization

how xpubs, taproot, and silent payments all work today
(classical)

secret key d

public key P = dG

generator point G

hash function H: $\{0,1\}^* \rightarrow Z$

$$\text{RerandomizeSecret}(d, \text{salt}) \rightarrow d' = d + H(P, \text{salt})$$

$$\begin{aligned}\text{RerandomizePublic}(P, \text{salt}) \rightarrow Q &= P + H(P, \text{salt}) \cdot G \\ &= d'G\end{aligned}$$

https://conduition.io

PQ rerandomization

xpubs, taproot, and silent payments after Q-day

what's possible?

hash-based signatures → impossible, no algebraic structure

lattice-based signatures → possible, but expensive (bigger sigs/pubkeys, slower).
best one I know of: https://cic.iacr.org/p/2/3/3/pdf
(|pk| + |sig| = 5429)

isogeny-based signatures? → feasible, opaque (no effect on sign/verify algos)

https://conduition.io

PQ rerandomization

xpubs, taproot, and silent payments after Q-day

with isogenies

RerandomizePublic

given a curve E_{pk}

derive a pseudorandom isogeny $\varphi : E_{\mathrm{pk}} \to E'_{\mathrm{pk}}$

using a hash $\varphi \leftarrow H(E_{\mathrm{pk}}, \mathrm{salt})$

return E'_{pk} as the new public key

fast, simple to implement

RerandomizeSecret

given $\mathrm{End}(E_{\mathrm{pk}})$

derive a pseudorandom isogeny $\varphi : E_{\mathrm{pk}} \to E'_{\mathrm{pk}}$

using a hash $\varphi \leftarrow H(E_{\mathrm{pk}}, \mathrm{salt})$

then find and return $\mathrm{End}(E'_{\mathrm{pk}})$ (e.g. using KLPT)

slower, more complex

https://conduition.io

recently proven secure!
https://eprint.iacr.org/2026/1169

review

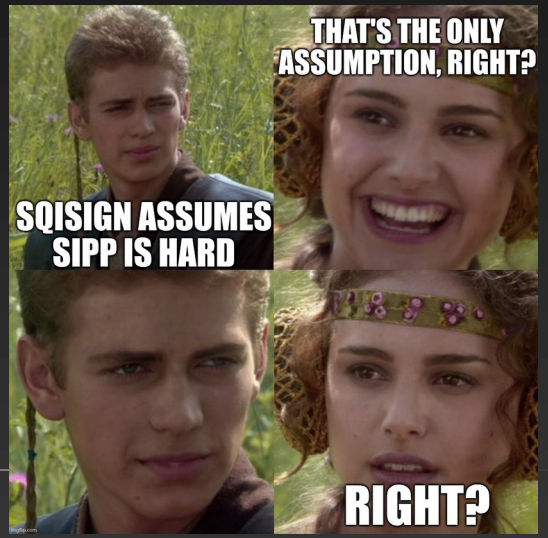
what did we even learn, anyway?

- isogenies are structure-preserving maps between elliptic curves
- forming a connected graph with supersingular curve iso-classes as the vertices
- supersingular curves are used as public keys
- isogeny signatures are a proof that you know the secret key End(E)
- those proofs are typically represented with isogenies (paths through the graph)
- signatures are compact because isogenies can be encoded as points
- security relies on the hardness of path-finding in the isogeny graph
- public keys can be rerandomized (BIP32) by pseudorandom graph walks

https://conduition.io

left unsaid

- quaternation algebras: maximal orders, ideals, lattices, etc
- performance: isogenies are pretty slow (see sqisign.org)
- Kani’s Lemma & higher-dimensional isogenies: bedrock of modern isogeny schemes
- PRISM: newer, simpler hash & sign scheme; less robust but simple & flexible
- additional security assumptions of SQIsign & PRISM (beyond SIPP/ERP)



sources

- Joseph Silverman, The Arithmetic of Elliptic Curves (2009)
- De Feo et al, SQISign: compact post-quantum signatures from quaternions and isogenies (2020)
- Basso et al, SQIsign2D-West: The Fast, the Small, and the Safer (2024)
- De Feo, Jao, & Plût, Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies (2011)
- Castryck & Decru, An efficient key recovery attack on SIDH (2022)
- Steven Galbraith, Mathematics of Public Key Cryptography, chapter 25 (2012)
- Basso et al, Isogeny-based Signatures with Randomizable Keys (2026)
- Basso et al, PRISM: Simple And Compact Identification and Signatures From Large Prime Degree Isogenies (2025)
- Herlédan Le Merdy & Wesolowski, Unconditional foundations for supersingular isogeny-based cryptography (2025)
- Aardal et al, A Complete Security Proof of SQIsign (2025)
- Luciano Maino, Isogeny-based Cryptography (undated)
- Luca De Feo, Mathematics of Isogeny Based Cryptography (2017)
- Damien Robert, On the efficient representation of isogenies (2024)

https://conduition.io

questions

https://conduition.io