docente: Nadia Fabrizio

Lab Blockchain (Addendum)

Fintech

April-May 2025

OUTLINE

1. Asymmetric KEY Exchange in Bitcoin
2. Ellliptic Curves
3. EC DSA

copyright © 2019 Cefriel – All rights reserved

Asymmetric Key Exchange- Private and Public Keys (Recall the concept)

Asymmetric cryptography is based on two keys associated to an identity:

- a **private** (secret) key *sk*, known only by the owner. It is generally used to encrypt a message by the owner,
  sk{m}

- a **public** key *pk* associated to the identity and public. It can decrypt a message encrypted with *sk*, i.e. ***pk{sk{m}} = m*** and also ***sk{pk{m]} = m***

Properties

- one way function: unfeasible from sk{m} to get m without pk, easy with pk

- with n nodes, only n pairs (pk_i,sk_i) are needed instead of n2

  anyone can secretly send to “j” by using pk_j{m}

- “j” can prove that they “own” a piece of information by using sk_j{m}

Signature based on Private and Public Keys

Asymmetric cryptography is based on two keys associated to an identity:

- a **private** (secret) key *sk*, known only by the owner. It is generally used to encrypt a message by the owner,***sk{m}***

- a **public** key *pk* associated to the identity and public. It can decrypt a message encrypted with *sk,* i.e. ***pk{sk{m}} = m*** and also ***sk{pk{m]} = m***

ES: Diffie Hellman, RSA….

Signature based on Private and Public Keys

Signature of a message (of a transaction) – on top of asymmetric cryptography:

- ***sig := sign(sk, message)***
- ***isValid := verify(pk, message, sig)***

By means of the public key *pk* one can validate the author of a message (**transaction**!)

How are they used in Bitcoin

You generate your own private key from a very large set, from there you derive your public key, and from the public key an **address** can be derived.

Bitcoin transactions move value from multiple input addresses to multiple output addresses.

Elliptic curves

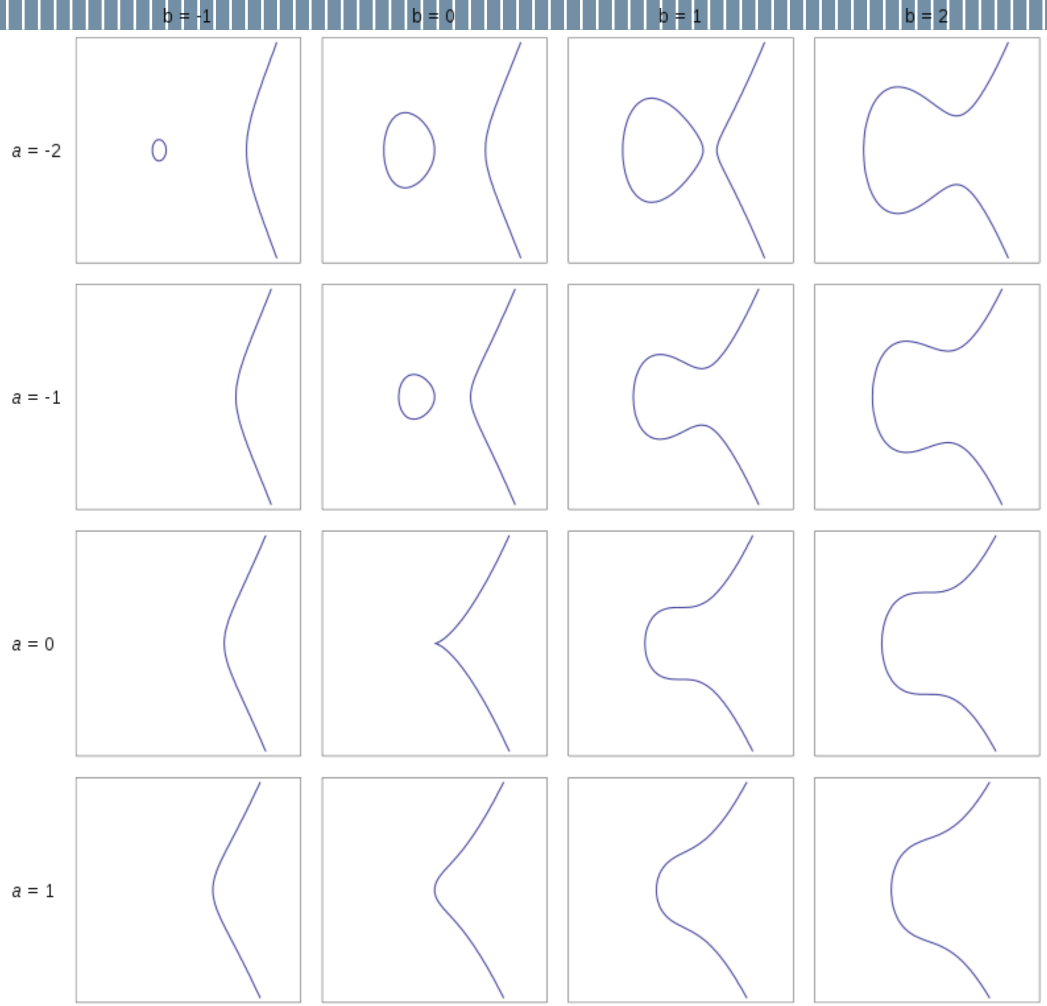
- In algebraic geometric, an **elliptic curves (EC) over $\mathbb{R}$ is** defined by the (*Weirstrass's equation*):
    - $$y^2 = x^3 + ax + b$$
- The curve is non singular if its determinant is non zero, which means
    - $$(-16 \quad 4a)^3 + 27b^2 \neq 0$$
- EC can be also defined over different field than $\mathbb{R}$, for instance over Finite Field (Galois)

copyright © 2019 Cefriel – All rights reserved

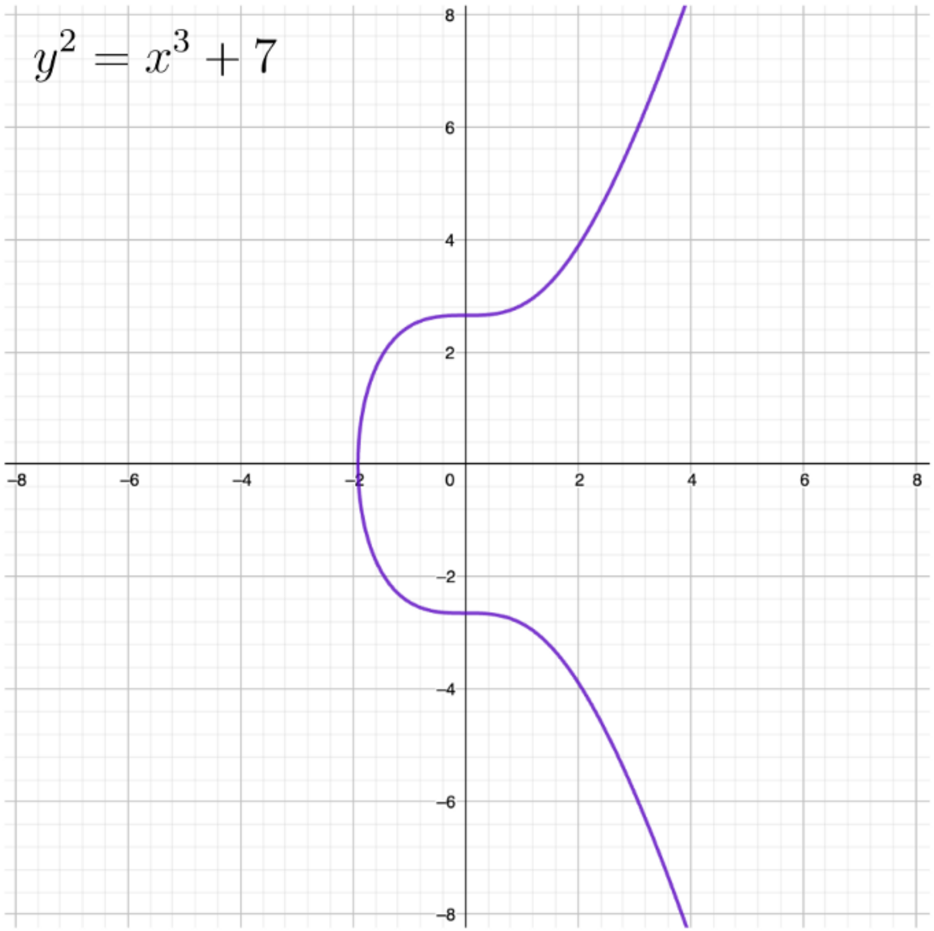
Elliptic C. properties

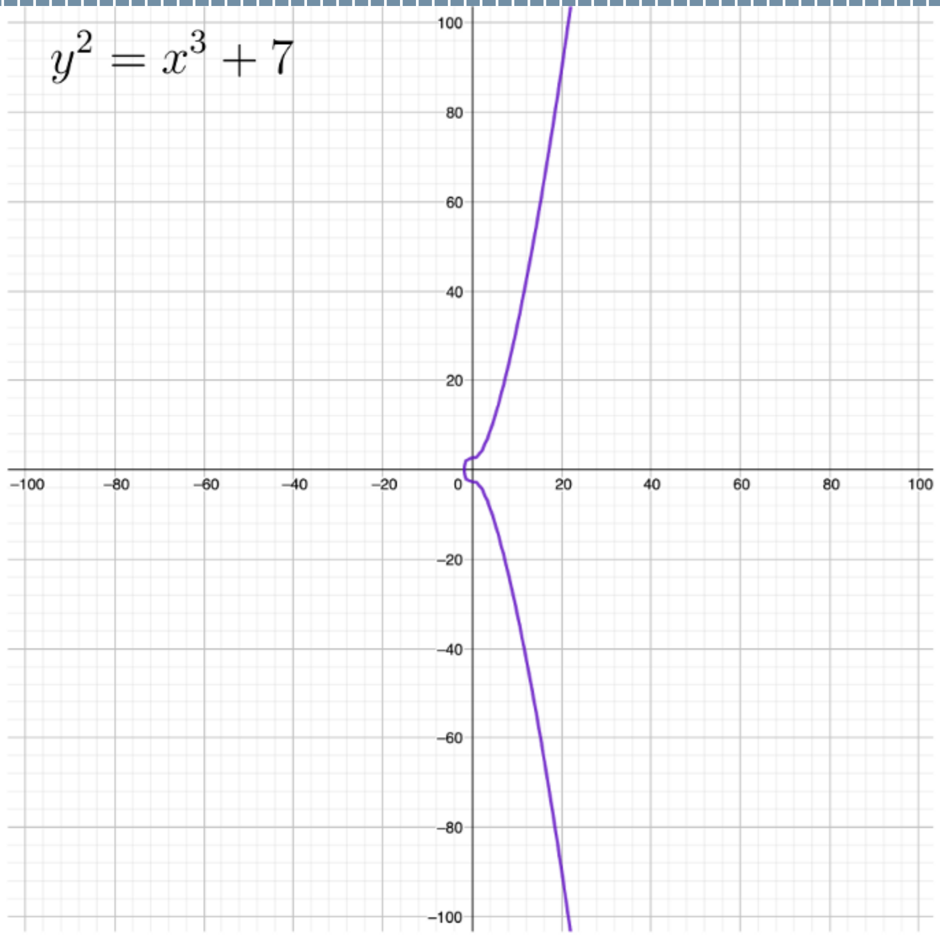
- It is possible to demonstrate that any elliptic curves is an Abelian Variety of dim 1
  - On any non singular EC , there exists a composition law
    (+) + the infinity closure
- In Bitcoin, it is used with a=-7 and b=10 over a Finite Field
  GF(263)

Elliptic CURVES

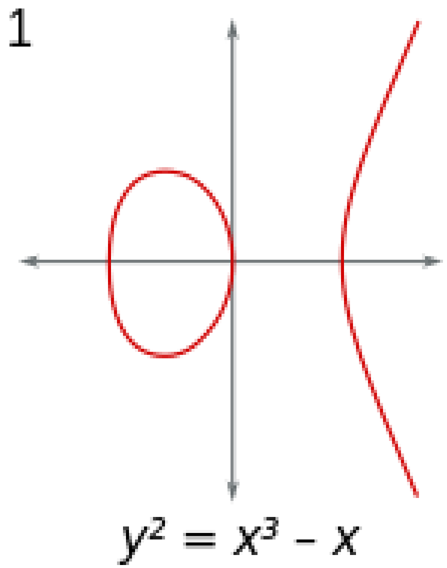


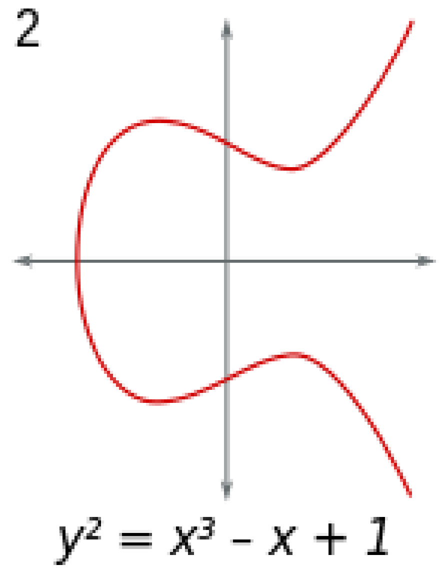
EXAMPLES

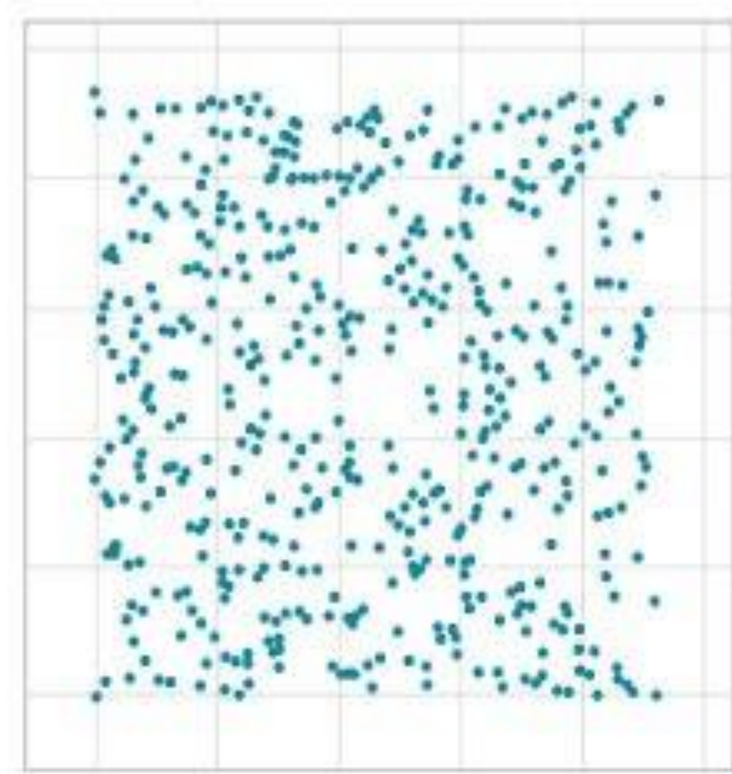




EC over R and over GF(p), where p is prime.



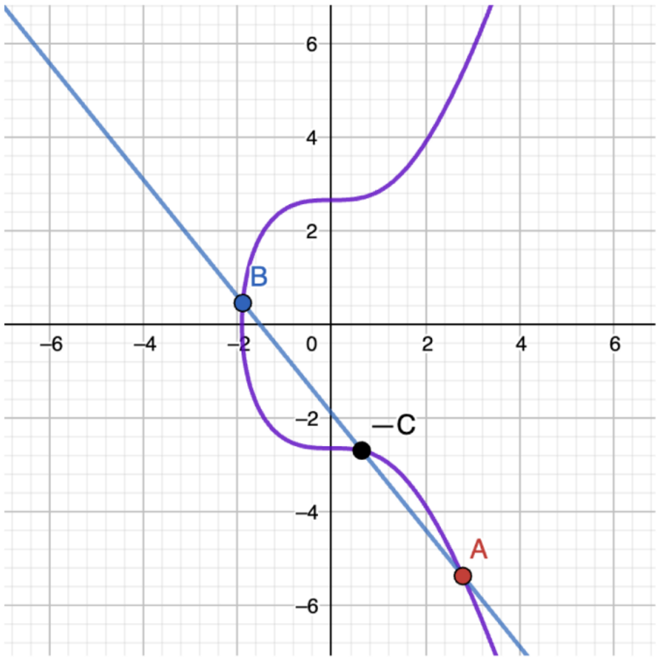


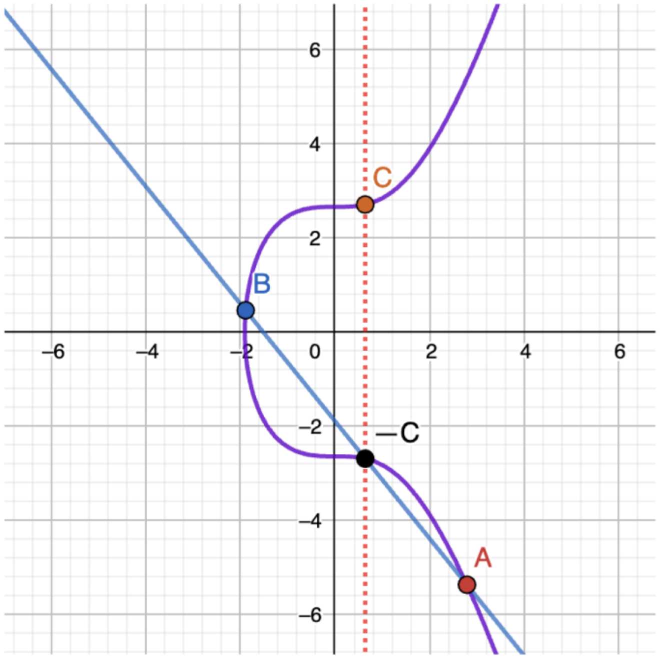


copyright © 2019 Cefriel – All rights reserved

https://medium.com/@blairlmarshall/how-does-ecdsa-work-in-bitcoin-7819d201a3ec

EC: properties-composition Law/SUM

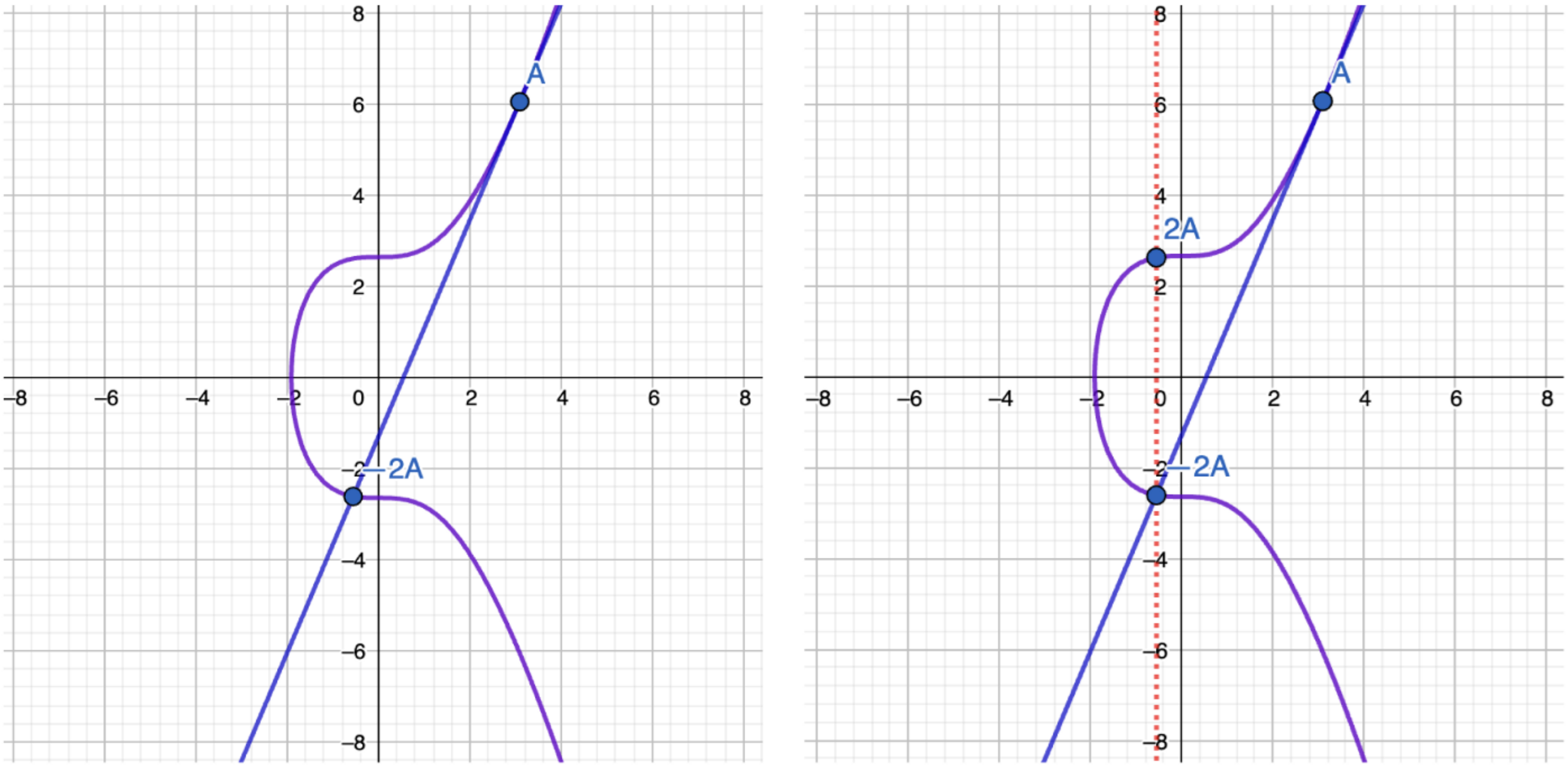




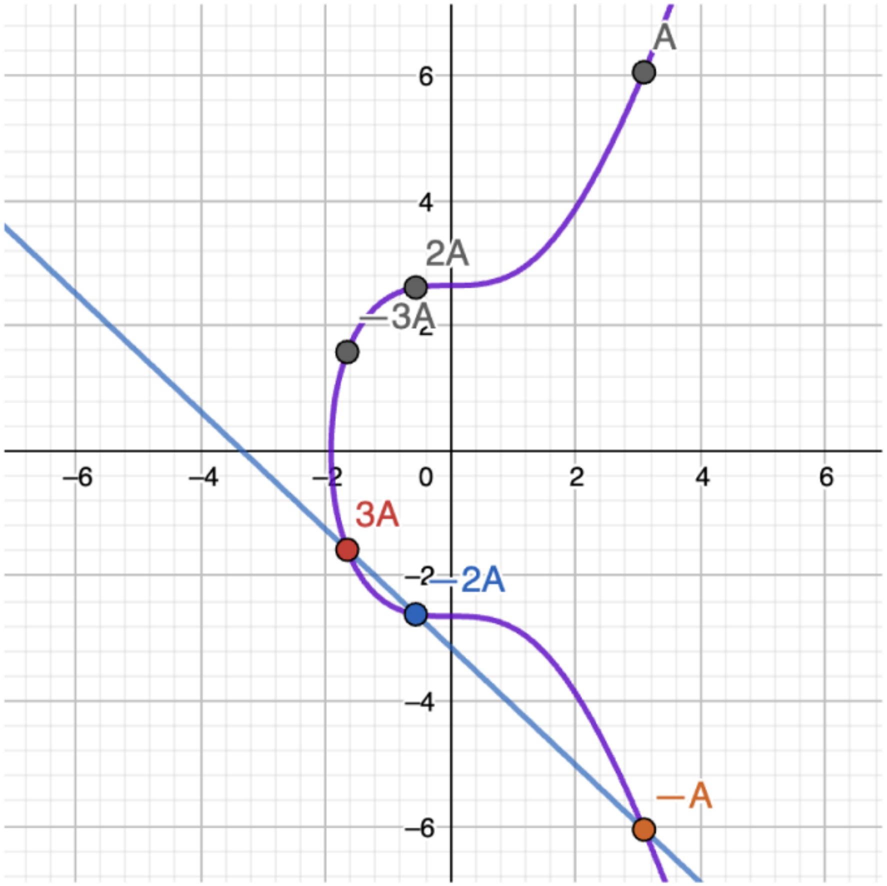
https://youtu.be/muIv8I6v1aE

C:= A+B

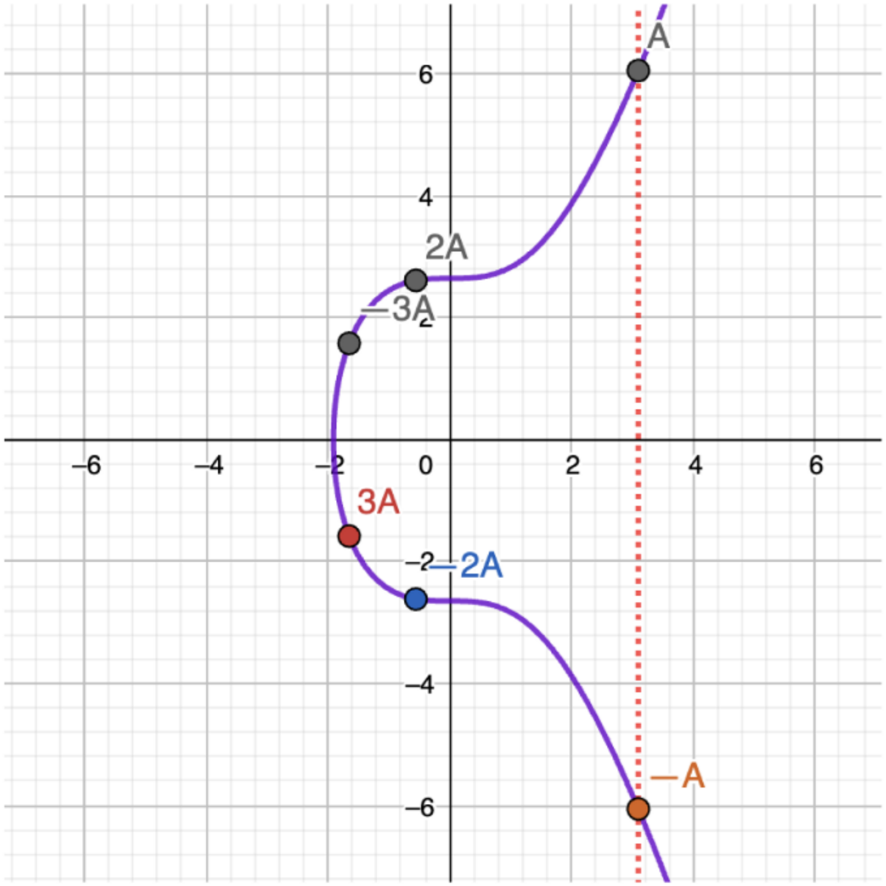
Multiply



....



A
2A
−3A
3A
−2A
−A



A
2A
−3A
3A
−2A
−A

For example

For example, to get 10A:

$2A = A + A$

$4A = 2A + 2A$

$8A = 4A + 4A$

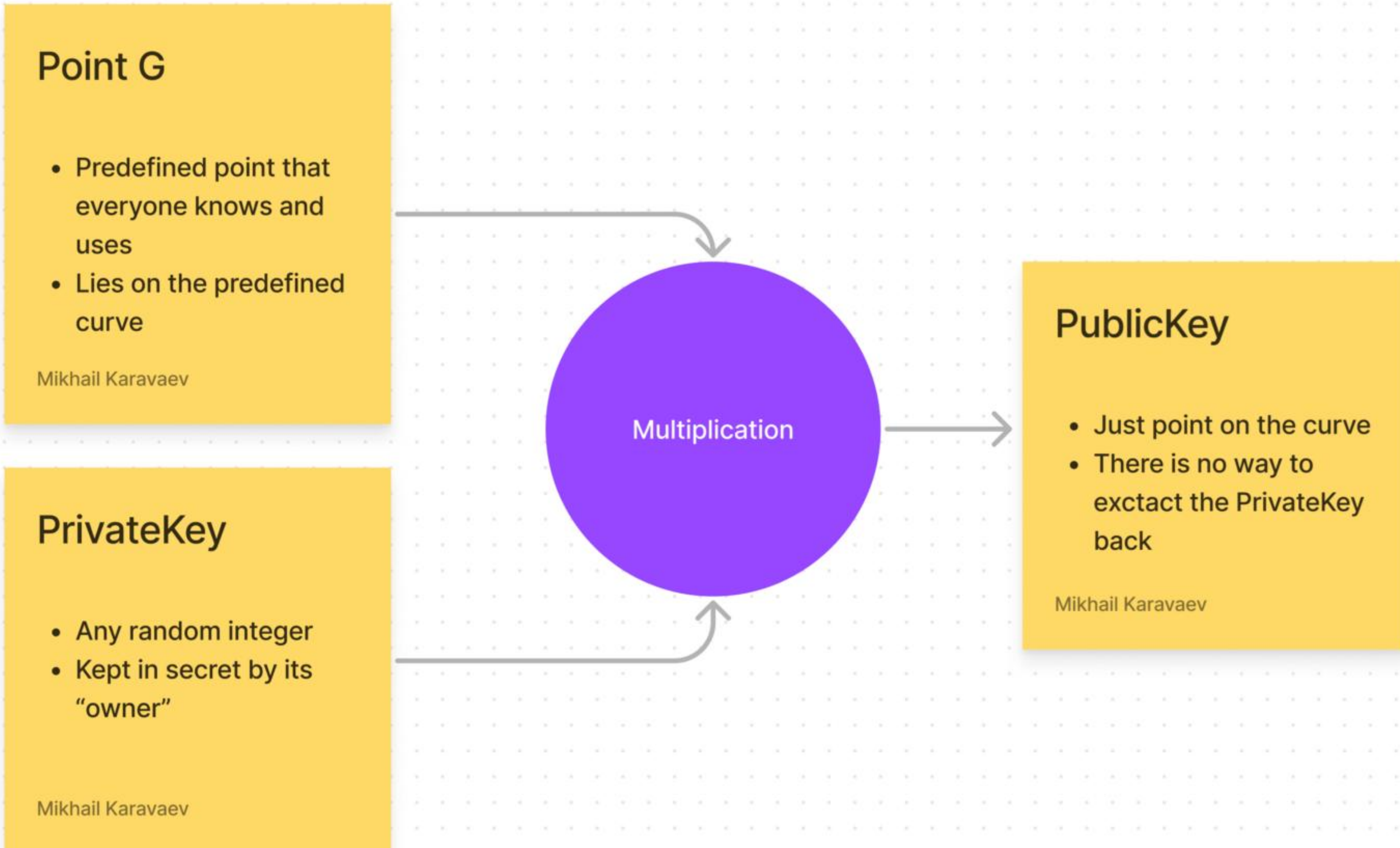
$10A = 8A + 2A$

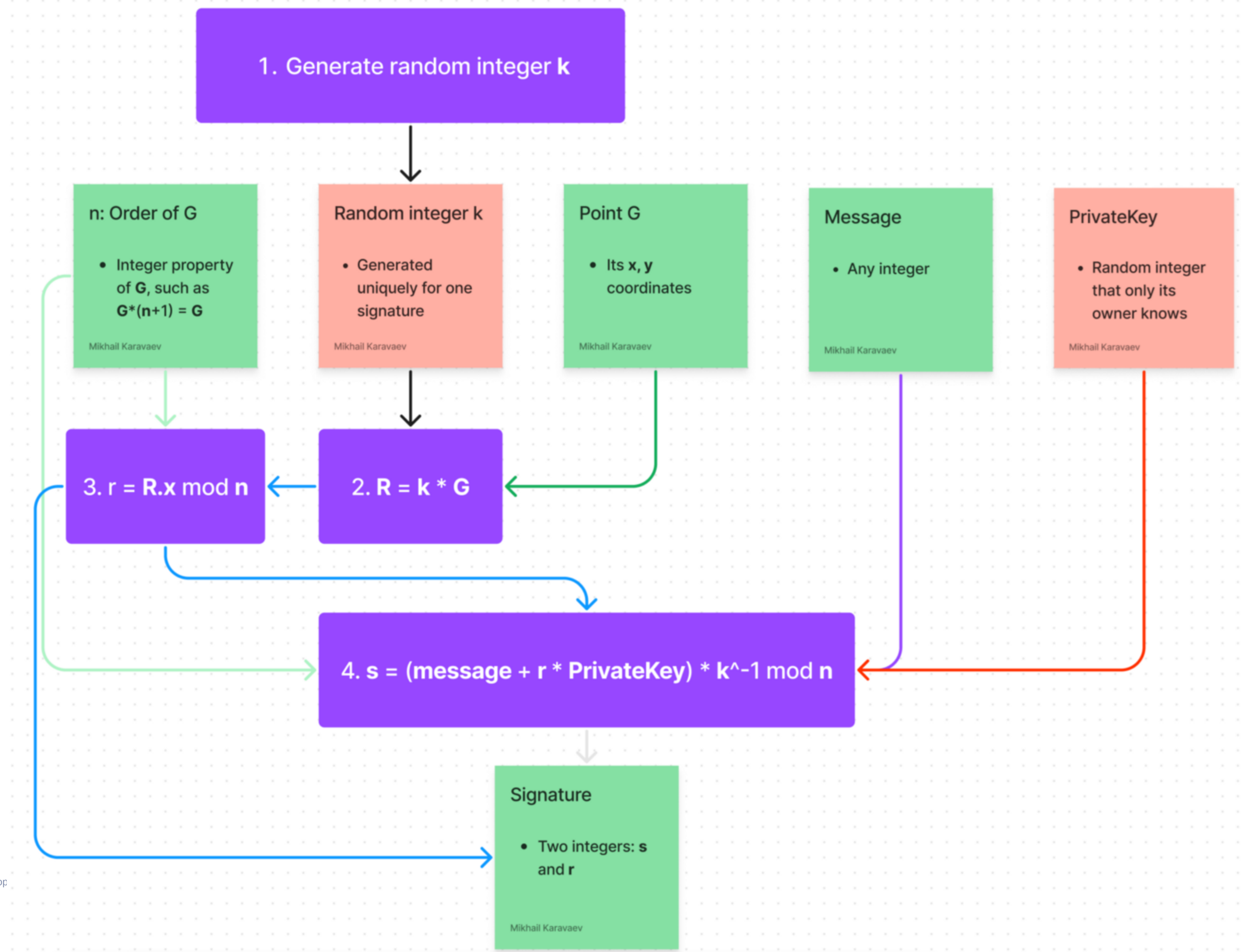
EC in Bitcoin: Secp256K1

- Bitcoin uses elliptic curves for public key
- It uses a famous EC, the Secp256k1
- C is over Fp (Koblitz curve secp256k1)
- EC is over Fp (Koblitz curve secp256k1)
  - where the finite field Fp is defined by:
    - p = FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFE FFFFFC2F= 2256 - 232 - 29 - 28 - 27 - 26 - 24 - 1, a=7, b=0
- As the b constant is zero, the ax term in the curve equation is always zero, hence the curve equation becomes y2 = x3 + 7.

defined in Standards for Efficient Cryptography (SEC) (Certicom Research, http://www.secg.org/sec2- v2.pdf).

y2 = x3 + 7.


Point G
Predefined point that everyone knows and uses
Lies on the predefined curve
Mikhail Karavaev
PrivateKey
Any random integer
Kept in secret by its "owner"
Mikhail Karavaev
Multiplication
PublicKey
Just point on the curve
There is no way to exctact the PrivateKey back
Mikhail Karavaev



1. Generate random integer k
n: Order of G
Integer property of G, such as G*(n+1) = G
Mikhail Karavaev
Random integer k
Generated uniquely for one signature
Mikhail Karavaev
Point G
Its x, y coordinates
Mikhail Karavaev
Message
Any integer
Mikhail Karavaev
PrivateKey
Random integer that only its owner knows
Mikhail Karavaev
3. r = R.x mod n
2. R = k * G
4. s = (message + r * PrivateKey) * k^-1 mod n
Signature
Two integers: s and r
Mikhail Karavaev

ADVANTAGES

Security Level (bits)	Ratio of DH Cost : EC Cost
80	3:1
112	6:1
128	10:1
192	32:1
256	64:1

Table 2: Relative Computation Costs of Diffie-Hellman and Elliptic Curves[1]

https://web.archive.org/web/20090117023500/http://www.nsa.gov/business/programs/elliptic_curve.shtml