Prime numbers and cryptography

Proseminar/Seminar

Dr. B. S. Banwait, C. V. Sriram

Seminar Vorbesprechung
Wednesday, 16^{th} February 2022

Interdisciplinary Center for Scientific Computing

UNIVERSITÄT HEIDELBERG
ZUKUNFT SEIT 1386

Introduction

Definition

A number p is prime if it has no proper divisors

Definition

A number p is prime if it has no proper divisors ,i.e.,

$$a|p \Rightarrow a = 1 \text{ or } p.$$

Definition

A number p is prime if it has no proper divisors ,i.e.,

$$a|p \Rightarrow a = 1 \text{ or } p.$$

Example

$2, 3, 5, 7, 11$ are primes.

Definition

A number p is prime if it has no proper divisors ,i.e.,

$$a|p \Rightarrow a = 1 \text{ or } p.$$

Example

$2, 3, 5, 7, 11$ are primes.

Their study goes back to the very beginnings of mathematics.

Rhind Mathematical Papyrus, from ca. 1550 BC, contains *Egyptian fraction* expansions of some prime numbers.

Proposition

Any integer is divisible by a prime number.

Proposition

Any integer is divisible by a prime number.

Proof.

If the integer N_0 is prime, we're done.

Proposition

Any integer is divisible by a prime number.

Proof.

If the integer N_0 is prime, we're done. If not, then there is an $N_1 | N_0$, $N_1 < N_0$. If N_1 is prime, we're done.

Proposition

Any integer is divisible by a prime number.

Proof.

If the integer N_0 is prime, we're done. If not, then there is an $N_1 | N_0$, $N_1 < N_0$. If N_1 is prime, we're done. Continue.

Proposition

Any integer is divisible by a prime number.

Proof.

If the integer N_0 is prime, we're done. If not, then there is an $N_1 | N_0$, $N_1 < N_0$. If N_1 is prime, we're done. Continue. If this didn't terminate, then you'd have an infinite decreasing sequence $N_0, N_1, \ldots$ of positive integers.

Proposition

Any integer is divisible by a prime number.

Proof.

If the integer N_0 is prime, we're done. If not, then there is an $N_1 | N_0$, $N_1 < N_0$. If N_1 is prime, we're done. Continue. If this didn't terminate, then you'd have an infinite decreasing sequence $N_0, N_1, \ldots$ of positive integers. Contradiction. $\square$

Theorem (Euclid, ca. 200 BC)

There are infinitely many prime numbers.

Euclid, from *The School of Athens* by Raphael, 1511

Theorem (Euclid, ca. 200 BC)

There are infinitely many prime numbers.

Euclid, from *The School of Athens* by Raphael, 1511

Proof.

Suppose not.

Theorem (Euclid, ca. 200 BC)

There are infinitely many prime numbers.

Euclid, from *The School of Athens* by Raphael, 1511

Proof.

Suppose not. Then list the primes in increasing order.

Theorem (Euclid, ca. 200 BC)

There are infinitely many prime numbers.

Euclid, from *The School of Athens* by Raphael, 1511

Proof.

Suppose not. Then list the primes in increasing order.

$$p_1, \ldots, p_n.$$

Consider the integer $p_1 \cdots p_n + 1$.

Theorem (Euclid, ca. 200 BC)

There are infinitely many prime numbers.

Euclid, from *The School of Athens* by Raphael, 1511

Proof.

Suppose not. Then list the primes in increasing order.

$$p_1, \dots, p_n.$$

Consider the integer $p_1 \cdots p_n + 1$. By the previous proposition, this must be divisible by a prime number, i.e., divisible by one of the p_i.

Theorem (Euclid, ca. 200 BC)

There are infinitely many prime numbers.

Euclid, from *The School of Athens* by Raphael, 1511

Proof.

Suppose not. Then list the primes in increasing order.

$$p_1, \ldots, p_n.$$

Consider the integer $p_1 \cdots p_n + 1$. By the previous proposition, this must be divisible by a prime number, i.e., divisible by one of the p_i. But it is not. Contradiction. $\square$

Prime numbers are the basis of cryptography

Cryptography has a very long history in mathematics.

Prime numbers are the basis of cryptography

Cryptography has a very long history in mathematics.

The first page of al-Kindi's manuscript *On Deciphering Cryptographic Messages*, ca 750 AD

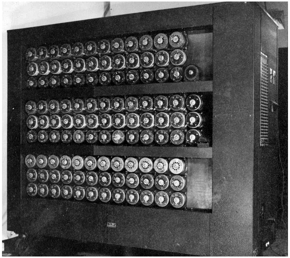
Also used in WW2 by both Allied and Axis powers.

Also used in WW2 by both Allied and Axis powers.

German *Enigma* machine for encryption.

Also used in WW2 by both Allied and Axis powers.

German *Enigma* machine for encryption.



British *Bombe* machine for decrypting Enigma messages. Designed by Alan Turing.

Today it gets used in

Today it gets used in

- The SSL/TLS protocol, and hence in `https`; as such it is heavily used for internet encryption;

Today it gets used in

- The SSL/TLS protocol, and hence in `https`; as such it is heavily used for internet encryption;
- The Signal protocol, used by messaging apps including Facebook Messenger, WhatsApp, and Signal;

Today it gets used in

- The SSL/TLS protocol, and hence in `https`; as such it is heavily used for internet encryption;
- The Signal protocol, used by messaging apps including Facebook Messenger, WhatsApp, and Signal;
- Verifying proof of ownership in the Bitcoin blockchain.

Today it gets used in

- The SSL/TLS protocol, and hence in `https`; as such it is heavily used for internet encryption;
- The Signal protocol, used by messaging apps including Facebook Messenger, WhatsApp, and Signal;
- Verifying proof of ownership in the Bitcoin blockchain.

The most modern cryptosystems are based on elliptic curves.

Rubric of the Proseminar talks

Talk 1 - Prime numbers and complexity analysis

- Basic properties, including the Fundamental theorem of arithmetic, that any integer can be expressed *uniquely* as a product of prime numbers:

$$N = p_1^{e_1} \cdots p_r^{e_r}.$$

Talk 1 - Prime numbers and complexity analysis

- Basic properties, including the Fundamental theorem of arithmetic, that any integer can be expressed *uniquely* as a product of prime numbers:
$$N = p_1^{e_1} \cdots p_r^{e_r}.$$
- Basics of modular arithmetic

Talk 1 - Prime numbers and complexity analysis

- Basic properties, including the Fundamental theorem of arithmetic, that any integer can be expressed *uniquely* as a product of prime numbers:

$$N = p_1^{e_1} \cdots p_r^{e_r}.$$

- Basics of modular arithmetic
- Quadratic reciprocity (statement) and Jacobi symbol computation

Talk 1 - Prime numbers and complexity analysis

- Basic properties, including the Fundamental theorem of arithmetic, that any integer can be expressed *uniquely* as a product of prime numbers:

$$N = p_1^{e_1} \cdots p_r^{e_r}.$$

- Basics of modular arithmetic
- Quadratic reciprocity (statement) and Jacobi symbol computation
- The prime number theorem, that there are "about" $N/\log(N)$ primes up to N.

Talk 1 - Prime numbers and complexity analysis

- Basic properties, including the Fundamental theorem of arithmetic, that any integer can be expressed *uniquely* as a product of prime numbers:

$$N = p_1^{e_1} \cdots p_r^{e_r}.$$

- Basics of modular arithmetic
- Quadratic reciprocity (statement) and Jacobi symbol computation
- The prime number theorem, that there are "about" $N/\log(N)$ primes up to N.
- The Riemann zeta function

$$\zeta(s) = \sum_{n \geq 1} \frac{1}{n^s},$$

Talk 1 - Prime numbers and complexity analysis

- Basic properties, including the Fundamental theorem of arithmetic, that any integer can be expressed *uniquely* as a product of prime numbers:
$$N = p_1^{e_1} \cdots p_r^{e_r}.$$
- Basics of modular arithmetic
- Quadratic reciprocity (statement) and Jacobi symbol computation
- The prime number theorem, that there are "about" $N/\log(N)$ primes up to N.
- The Riemann zeta function
$$\zeta(s) = \sum_{n \geq 1} \frac{1}{n^s},$$
and its Euler product decomposition
$$\zeta(s) = \prod_p \frac{1}{1 - p^{-s}}.$$

Talks 2+3 - Fast arithmetic

The basic question for these talks is:

Talks 2+3 - Fast arithmetic

The basic question for these talks is:

Question

How can you quickly multiply very large numbers together?

Talks 2+3 - Fast arithmetic

The basic question for these talks is:

Question

How can you quickly multiply very large numbers together? Or divide them?

Talks 2+3 - Fast arithmetic

The basic question for these talks is:

Question

How can you quickly multiply very large numbers together? Or divide them? Or take square roots?

Talks 2+3 - Fast arithmetic

The basic question for these talks is:

Question

How can you quickly multiply very large numbers together? Or divide them? Or take square roots?

These basic arithmetic operations are of fundamental importance for computers and embedded processor design.

Talks 2+3 - Fast arithmetic

The basic question for these talks is:

Question

How can you quickly multiply very large numbers together? Or divide them? Or take square roots?

These basic arithmetic operations are of fundamental importance for computers and embedded processor design.

These talks will give an overview of some of these fast methods.

Talk 4 - Pseudoprimes and the Miller-Rabin test

Recall Fermat's Little theorem:

Talk 4 - Pseudoprimes and the Miller-Rabin test

Recall Fermat's Little theorem:

Theorem

For p a prime and a coprime to p, we have

$$p | a^{p-1} - 1.$$

Talk 4 - Pseudoprimes and the Miller-Rabin test

Recall Fermat's Little theorem:

Theorem

For p a prime and a coprime to p, we have

$$p | a^{p-1} - 1.$$

The converse is not true; there are composite numbers N such that, for some a coprime to N,

$$N | a^{N-1} - 1.$$

Talk 4 - Pseudoprimes and the Miller-Rabin test

Recall Fermat's Little theorem:

Theorem

For p a prime and a coprime to p, we have

$$p | a^{p-1} - 1.$$

The converse is not true; there are composite numbers N such that, for some a coprime to N,

$$N | a^{N-1} - 1.$$

Such N are called Fermat pseudoprimes to base a.

Talk 4 - Pseudoprimes and the Miller-Rabin test

Recall Fermat's Little theorem:

Theorem

For p a prime and a coprime to p, we have

$$p | a^{p-1} - 1.$$

The converse is not true; there are composite numbers N such that, for some a coprime to N,

$$N | a^{N-1} - 1.$$

Such N are called Fermat pseudoprimes to base a. This talk will show that there are infinitely many such N.

Talk 4 - Pseudoprimes and the Miller-Rabin test

Recall Fermat's Little theorem:

Theorem

For p a prime and a coprime to p, we have

$$p | a^{p-1} - 1.$$

The converse is not true; there are composite numbers N such that, for some a coprime to N,

$$N | a^{N-1} - 1.$$

Such N are called Fermat pseudoprimes to base a. This talk will show that there are infinitely many such N. If N satisfies the above divisibility *for all a coprime to N*, then N is called a Carmichael number.

Talk 4 - Pseudoprimes and the Miller-Rabin test

Recall Fermat's Little theorem:

Theorem

For p a prime and a coprime to p, we have

$$p | a^{p-1} - 1.$$

The converse is not true; there are composite numbers N such that, for some a coprime to N,

$$N | a^{N-1} - 1.$$

Such N are called Fermat pseudoprimes to base a. This talk will show that there are infinitely many such N. If N satisfies the above divisibility *for all a coprime to N*, then N is called a Carmichael number. These will also be studied.

Talk 4 - Pseudoprimes and the Miller-Rabin test

Recall Fermat's Little theorem:

Theorem

For p a prime and a coprime to p, we have

$$p | a^{p-1} - 1.$$

The converse is not true; there are composite numbers N such that, for some a coprime to N,

$$N | a^{N-1} - 1.$$

Such N are called Fermat pseudoprimes to base a. This talk will show that there are infinitely many such N. If N satisfies the above divisibility *for all a coprime to N*, then N is called a Carmichael number. These will also be studied. Also included in this talk will be the Miller-Rabin primality test, a probabilistic algorithm to determine primality.

Talks 5+6 - The Agrawal-Kayal-Saxena theorem

This important theorem from 2002 gives the first deterministic and unconditional algorithm to determine whether a number is prime in polynomial time.

Talks 5+6 - The Agrawal-Kayal-Saxena theorem

This important theorem from 2002 gives the first deterministic and unconditional algorithm to determine whether a number is prime in polynomial time.

Input: integer $n > 1$.

1. Check if n is a perfect power: if $n = a^b$ for integers $a > 1$ and $b > 1$, output *composite*.
2. Find the smallest r such that $\mathrm{ord}_r(n) > (\log_2 n)^2$. (if r and n are not coprime, then skip this r)
3. For all $2 \le a \le \min(r, n-1)$, check that a does not divide n: If $a|n$ for some $2 \le a \le \min(r, n-1)$, output *composite*.
4. If $n \le r$, output *prime*.
5. For $a = 1$ to $\left\lfloor \sqrt{\varphi(r)} \log_2(n) \right\rfloor$ do

   if $(X+a)^n \neq X^n + a \pmod{X^r - 1, n}$, output *composite*;
6. Output *prime*.

Talks 5+6 - The Agrawal-Kayal-Saxena theorem

This important theorem from 2002 gives the first deterministic and unconditional algorithm to determine whether a number is prime in polynomial time.

Input: integer $n > 1$.

1. Check if n is a perfect power: if $n = a^b$ for integers $a > 1$ and $b > 1$, output *composite*.
2. Find the smallest r such that $\mathrm{ord}_r(n) > (\log_2 n)^2$. (if r and n are not coprime, then skip this r)
3. For all $2 \le a \le \min(r, n-1)$, check that a does not divide n: If $a | n$ for some $2 \le a \le \min(r, n-1)$, output *composite*.
4. If $n \le r$, output *prime*.
5. For $a = 1$ to $\left\lfloor \sqrt{\varphi(r)} \log_2(n) \right\rfloor$ do

   if $(X+a)^n \neq X^n + a \pmod{X^r - 1, n}$, output *composite*;
6. Output *prime*.

These two talks will present the details and consequences.

Talks 7+8 - Factorisation methods

These two talks are about factoring integers.

Talks 7+8 - Factorisation methods

These two talks are about factoring integers.

- Pollard's ρ-method

Talks 7+8 - Factorisation methods

These two talks are about factoring integers.

- Pollard's ρ-method
- The discrete logarithm problem

Talks 7+8 - Factorisation methods

These two talks are about factoring integers.

- Pollard's ρ-method
- The discrete logarithm problem
- The Baby-Step-Giant-Step method

Talks 7+8 - Factorisation methods

These two talks are about factoring integers.

- Pollard's ρ-method
- The discrete logarithm problem
- The Baby-Step-Giant-Step method
- Smooth numbers and the Sieve of Eratosthenes

Talks 7+8 - Factorisation methods

These two talks are about factoring integers.

- Pollard's ρ-method
- The discrete logarithm problem
- The Baby-Step-Giant-Step method
- Smooth numbers and the Sieve of Eratosthenes
- The quadratic sieve

Talks 7+8 - Factorisation methods

These two talks are about factoring integers.

- Pollard's ρ-method
- The discrete logarithm problem
- The Baby-Step-Giant-Step method
- Smooth numbers and the Sieve of Eratosthenes
- The quadratic sieve

Sieving refers to progressively removing composite numbers up to a bound, leaving only the primes behind.

Rubric of the Proseminar or Seminar Talks

Talk 9 - Overview of Algebraic number theory

Algebraic number theory is about replacing the standard integers $\mathbb{Z}$ with more general number rings. E.g.

Talk 9 - Overview of Algebraic number theory

Algebraic number theory is about replacing the standard integers $\mathbb{Z}$ with more general number rings. E.g.

$$\mathbb{Z}[i] = \{a + ib : a, b \in \mathbb{Z}\}.$$

It turns out that the Fundamental theorem of arithmetic does not necessarily hold in these more general number rings. E.g.

Talk 9 - Overview of Algebraic number theory

Algebraic number theory is about replacing the standard integers $\mathbb{Z}$ with more general number rings. E.g.

$$\mathbb{Z}[i] = \{a + ib : a, b \in \mathbb{Z}\} .$$

It turns out that the Fundamental theorem of arithmetic does not necessarily hold in these more general number rings. E.g. in

$$\mathbb{Z}[\sqrt{-5}] = \left\{a + b\sqrt{-5} : a, b \in \mathbb{Z}\right\}$$

we have

Talk 9 - Overview of Algebraic number theory

Algebraic number theory is about replacing the standard integers $\mathbb{Z}$ with more general number rings. E.g.

$$\mathbb{Z}[i] = \{a + ib : a, b \in \mathbb{Z}\}.$$

It turns out that the Fundamental theorem of arithmetic does not necessarily hold in these more general number rings. E.g. in

$$\mathbb{Z}[\sqrt{-5}] = \left\{a + b\sqrt{-5} : a, b \in \mathbb{Z}\right\}$$

we have

$$6 = 2 \times 3 = (1 + \sqrt{-5}) \times (1 - \sqrt{-5}),$$

Talk 9 - Overview of Algebraic number theory

Algebraic number theory is about replacing the standard integers $\mathbb{Z}$ with more general number rings. E.g.

$$\mathbb{Z}[i] = \{a + ib : a, b \in \mathbb{Z}\}.$$

It turns out that the Fundamental theorem of arithmetic does not necessarily hold in these more general number rings. E.g. in

$$\mathbb{Z}[\sqrt{-5}] = \left\{a + b\sqrt{-5} : a, b \in \mathbb{Z}\right\}$$

we have

$$6 = 2 \times 3 = (1 + \sqrt{-5}) \times (1 - \sqrt{-5}),$$

and all of 2, 3, $(1 \pm \sqrt{-5})$ are all "prime" (actually, *irreducible*).

Talk 9 - Overview of Algebraic number theory

Algebraic number theory is about replacing the standard integers $\mathbb{Z}$ with more general number rings. E.g.

$$\mathbb{Z}[i] = \{a + ib : a, b \in \mathbb{Z}\}.$$

It turns out that the Fundamental theorem of arithmetic does not necessarily hold in these more general number rings. E.g. in

$$\mathbb{Z}[\sqrt{-5}] = \left\{a + b\sqrt{-5} : a, b \in \mathbb{Z}\right\}$$

we have

$$6 = 2 \times 3 = (1 + \sqrt{-5}) \times (1 - \sqrt{-5}),$$

and all of 2, 3, $(1 \pm \sqrt{-5})$ are all "prime" (actually, *irreducible*). This talk will give an overview of these key ideas.

Talk 10 - The number field sieve

This is the most efficient classical algorithm for factoring integers larger than 10^{100}, and requires ideas from algebraic number theory for its construction.

Talk 10 - The number field sieve

This is the most efficient classical algorithm for factoring integers larger than 10^{100}, and requires ideas from algebraic number theory for its construction.

- Construction and overview of the basic number field sieve

Talk 10 - The number field sieve

This is the most efficient classical algorithm for factoring integers larger than 10^{100}, and requires ideas from algebraic number theory for its construction.

- Construction and overview of the basic number field sieve
- Complexity

Talk 10 - The number field sieve

This is the most efficient classical algorithm for factoring integers larger than 10^{100}, and requires ideas from algebraic number theory for its construction.

- Construction and overview of the basic number field sieve
- Complexity
- Relation to the quadratic sieve

Talk 11 - Brief overview of elliptic curves

Elliptic curves are equations of the form

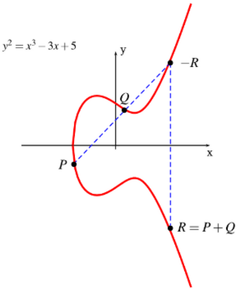
$$y^2 = x^3 + Ax + B.$$

Talk 11 - Brief overview of elliptic curves

Elliptic curves are equations of the form

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

They are remarkable because their set of solutions form a group under the chord and tangent process:

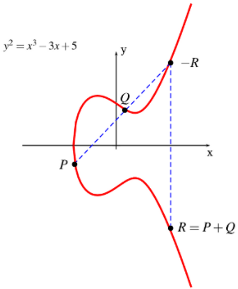


Talk 11 - Brief overview of elliptic curves

Elliptic curves are equations of the form

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

They are remarkable because their set of solutions form a group under the chord and tangent process:

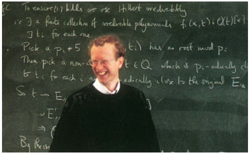


The identity of the group law is the point at infinity O_E, infinitely far up the y-axis.

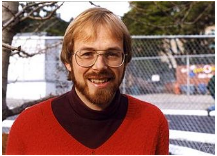
Modularity Theorem (Wiles, Taylor-Wiles, 1995)

Every elliptic curve $E/\mathbb{Q}$ arises from a (weight-2 cuspidal of level $\Gamma_0(N)$) modular form f_E such that the L-functions coincide:

$$L(E,s) = L(f_E,s).$$



Andrew J. Wiles



Richard L. Taylor

Modularity Theorem (Wiles, Taylor-Wiles, 1995)

Every elliptic curve $E/\mathbb{Q}$ arises from a (weight-2 cuspidal of level $\Gamma_0(N)$) modular form f_E such that the L-functions coincide:

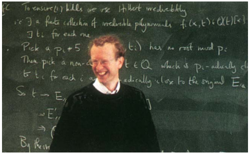
$$L(E, s) = L(f_E, s).$$

Corollary

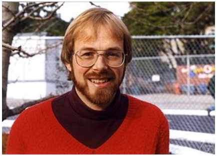
Fermat's Last Theorem is true: for $n > 2$, the equation

$$x^n + y^n = z^n$$

admits only the trivial solutions (i.e. when $xyz = 0$).



Andrew J. Wiles



Richard L. Taylor

They are also the subject of the Birch-Swinnerton-Dyer conjecture, one of the Clay Millenium Problems - solving it will earn you $1,000,000!

H.P.F. Swinnerton-Dyer with B. Birch

Talk 12 - Public-key cryptography

Talk 12 - Public-key cryptography

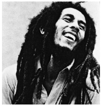
Elliptic curves over finite fields are also used extensively in public-key cryptography, via the Elliptic Curve Diffie-Hellman (ECDH) protocol:

Talk 12 - Public-key cryptography

Elliptic curves over finite fields are also used extensively in public-key cryptography, via the Elliptic Curve Diffie-Hellman (ECDH) protocol:

Alice

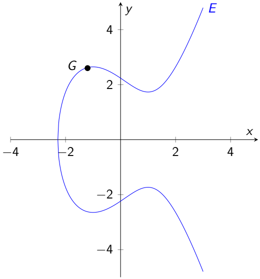
Bob



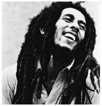
Talk 12 - Public-key cryptography

Elliptic curves over finite fields are also used extensively in public-key cryptography, via the Elliptic Curve Diffie-Hellman (ECDH) protocol:

Alice



Bob



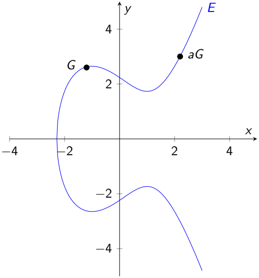
Domain parameters:

$$E, G \in E$$

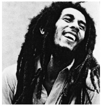
Talk 12 - Public-key cryptography

Elliptic curves over finite fields are also used extensively in public-key cryptography, via the Elliptic Curve Diffie-Hellman (ECDH) protocol:

Alice



Bob



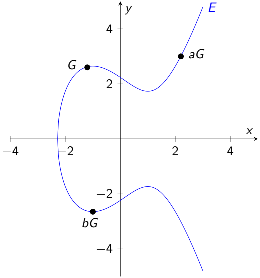
Private key: $a \in \mathbb{Z}$
Public key: $a \cdot G$

Domain parameters:
$E, G \in E$

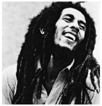
Talk 12 - Public-key cryptography

Elliptic curves over finite fields are also used extensively in public-key cryptography, via the Elliptic Curve Diffie-Hellman (ECDH) protocol:

Alice



Bob



Private key: $a \in \mathbb{Z}$
Public key: $a \cdot G$

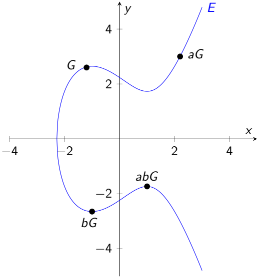
Domain parameters:
$E, G \in E$

Private key: $b \in \mathbb{Z}$
Public key: $b \cdot G$

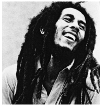
Talk 12 - Public-key cryptography

Elliptic curves over finite fields are also used extensively in public-key cryptography, via the Elliptic Curve Diffie-Hellman (ECDH) protocol:

Alice



Bob



Private key: $a \in \mathbb{Z}$
Public key: $a \cdot G$

Domain parameters:
$E, G \in E$

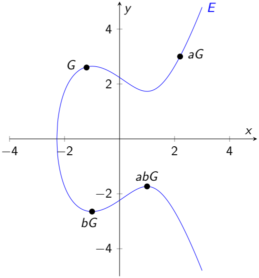
Private key: $b \in \mathbb{Z}$
Public key: $b \cdot G$

Shared Secret $= abG = baG$

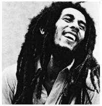
Talk 12 - Public-key cryptography

Elliptic curves over finite fields are also used extensively in public-key cryptography, via the Elliptic Curve Diffie-Hellman (ECDH) protocol:

Alice



Bob



Private key: $a \in \mathbb{Z}$
Public key: $a \cdot G$

Domain parameters:
$E, G \in E$

Private key: $b \in \mathbb{Z}$
Public key: $b \cdot G$

Shared Secret $= abG = baG$

This works because computing a from aG and G is a computationally infeasible problem!

Talk 13 - Factoring numbers with elliptic curves

In 1987 Hendrik W. Lenstra discovered a surprising method of factoring numbers using elliptic curves.

Hendrik W. Lenstra

Talk 14 - Post-quantum cryptogprahy, and SIDH

Talk 14 - Post-quantum cryptogprahy, and SIDH

A very active area in cryptography is aimed at developing cryptographic primitives which would be secure against quantum-computational attack.

Talk 14 - Post-quantum cryptogprahy, and SIDH

A very active area in cryptography is aimed at developing cryptographic primitives which would be secure against quantum-computational attack.

IBM Q System One

Talk 14 - Post-quantum cryptogprahy, and SIDH

A very active area in cryptography is aimed at developing cryptographic primitives which would be secure against quantum-computational attack.

The United States' National Institute of Standards and Technology currently has an ongoing competition to select primitives that would be recommended for widespread use.

IBM Q System One

NIST National Institute of Standards and Technology U.S. Department of Commerce

Talk 14 - Post-quantum cryptogprahy, and SIDH

A very active area in cryptography is aimed at developing cryptographic primitives which would be secure against quantum-computational attack.

The United States' National Institute of Standards and Technology currently has an ongoing competition to select primitives that would be recommended for widespread use.

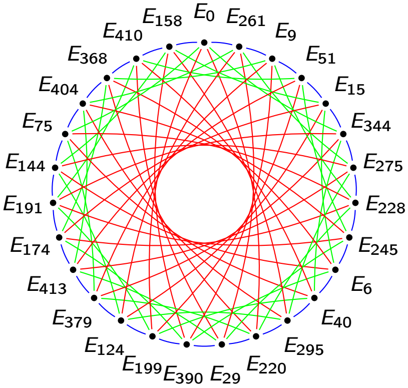
IBM Q System One

NIST National Institute of Standards and Technology U.S. Department of Commerce

One primitive which has reached Round 3 of the competition - Supersingular Isogeny Key Encapsulation - is based on the theory of isogenies of elliptic curves.

Isogeny-based cryptography

The domain parameter consists of a supersingular isogeny graph of elliptic curves over a finite field:



Nodes: Supersingular elliptic curves $E_A : y^2 = x^3 + Ax^2 + x$ over $\mathbb{F}_{419}$.
Edges: 3-, 5-, and 7-isogenies (more details to come.)

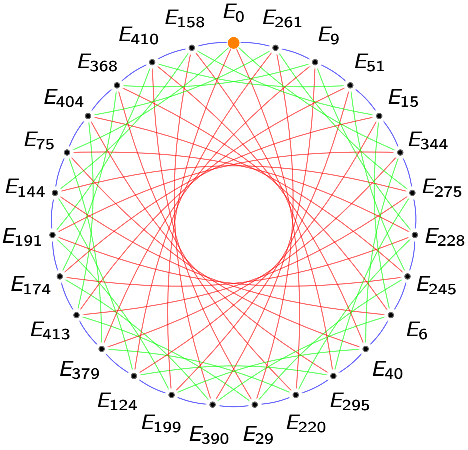
Diffie-Hellman on supersingular isogeny graphs

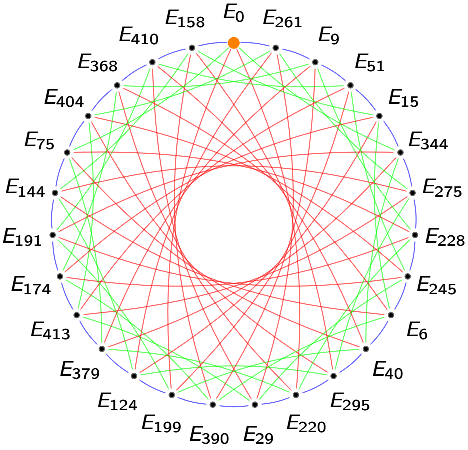
Alice

$a = [+, -, +, -]$

Bob

$b = [+, +, -, +]$



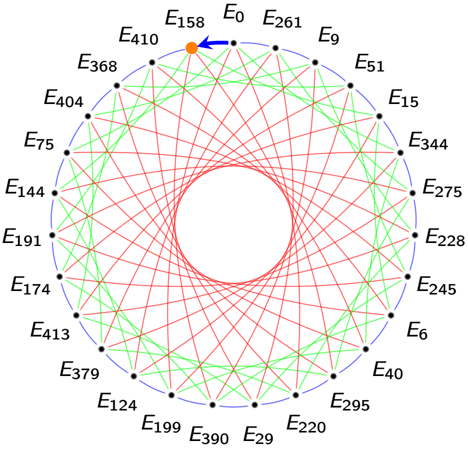


Diffie-Hellman on supersingular isogeny graphs

Alice

$a = [{\color{blue}+}, {\color{green}-}, {\color{blue}+}, {\color{red}-}]$

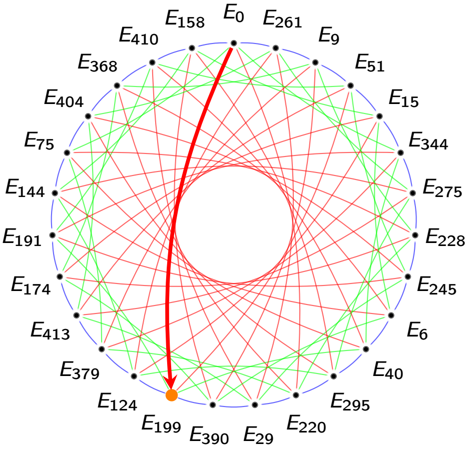
$\uparrow$



Bob

$b = [{\color{red}+}, {\color{green}+}, {\color{blue}-}, {\color{red}+}]$

$\uparrow$

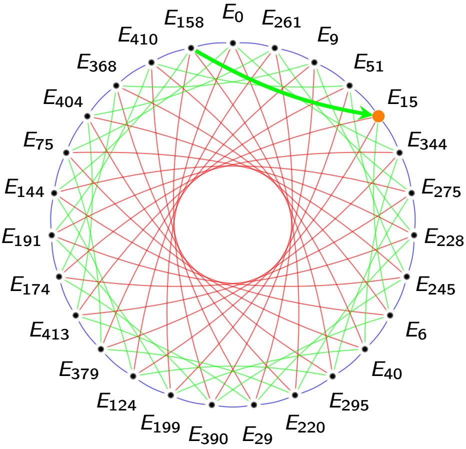


Diffie-Hellman on supersingular isogeny graphs

Alice

$a = [+, -, +, -]$

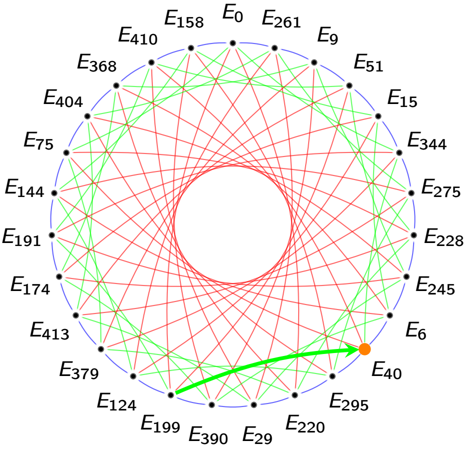
$\uparrow$



Bob

$b = [+, +, -, +]$

$\uparrow$

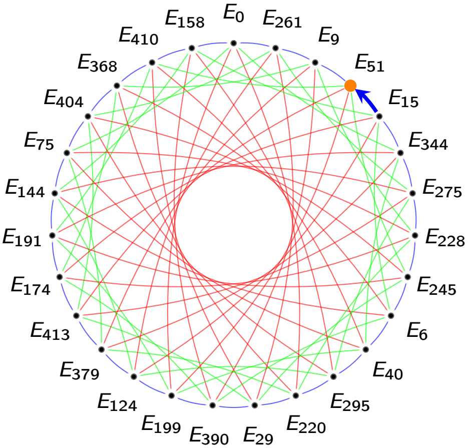


Diffie-Hellman on supersingular isogeny graphs

Alice

$a = [+, -, +, -]$

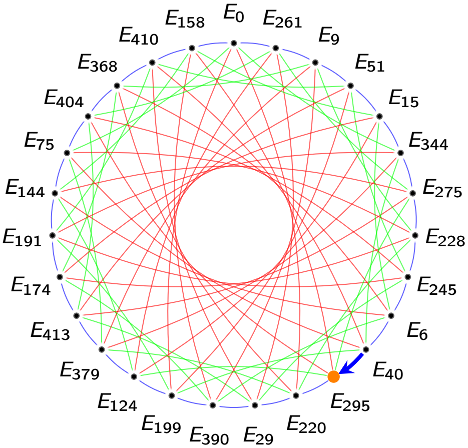
$\uparrow$



Bob

$b = [+, +, -, +]$

$\uparrow$

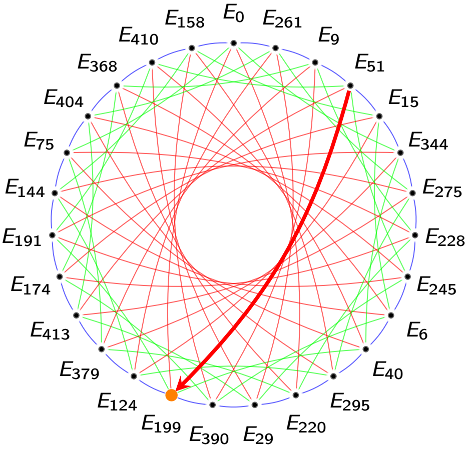


Diffie-Hellman on supersingular isogeny graphs

Alice

$a = [+, -, +, -]$

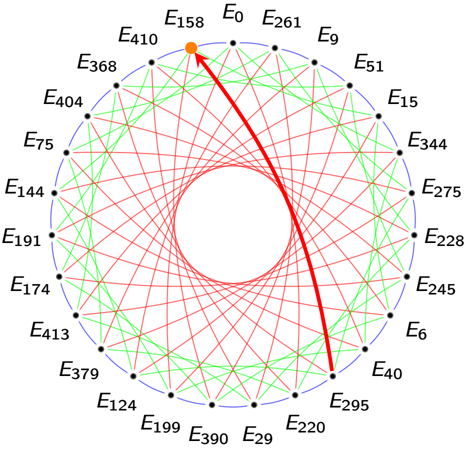
$\uparrow$



Bob

$b = [+, +, -, +]$

$\uparrow$



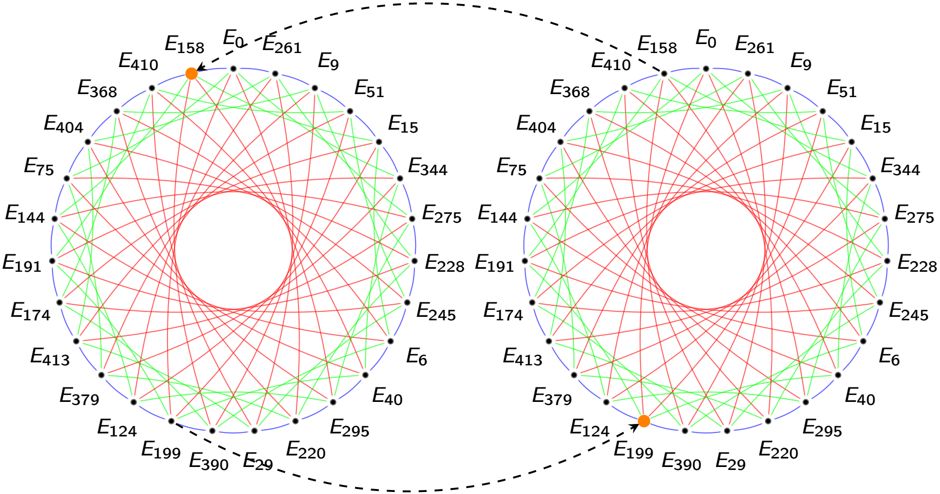
Diffie-Hellman on supersingular isogeny graphs

Alice

$a = [+, -, +, -]$

Bob

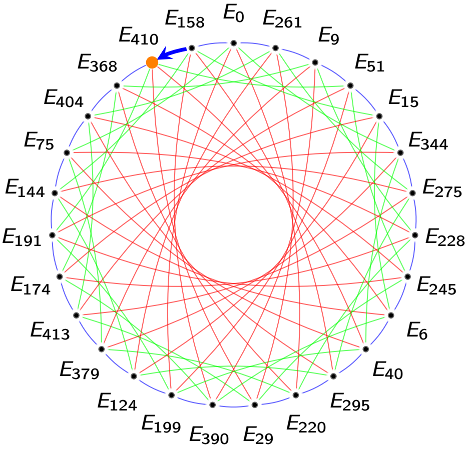
$b = [+, +, -, +]$



Diffie-Hellman on supersingular isogeny graphs

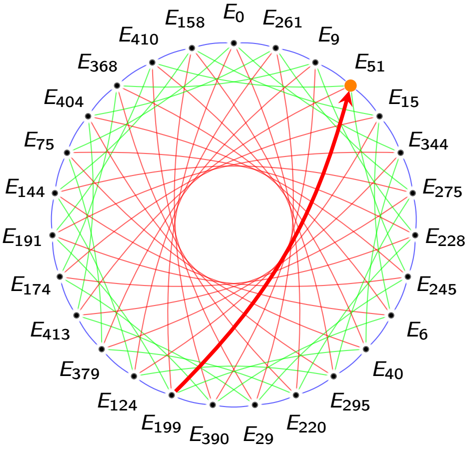
Alice

$$a = [+, -, +, -]$$
$$\uparrow$$



Bob

$$b = [+, +, -, +]$$
$$\uparrow$$

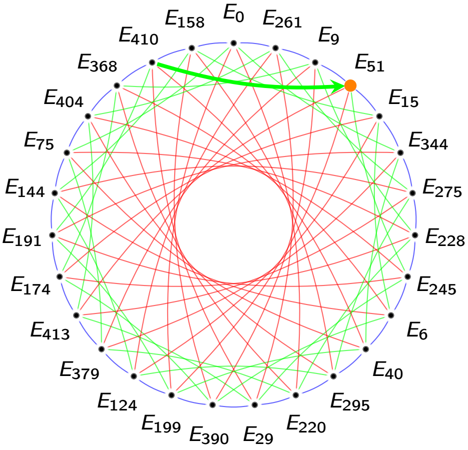


Diffie-Hellman on supersingular isogeny graphs

Alice

$a = [+, -, +, -]$

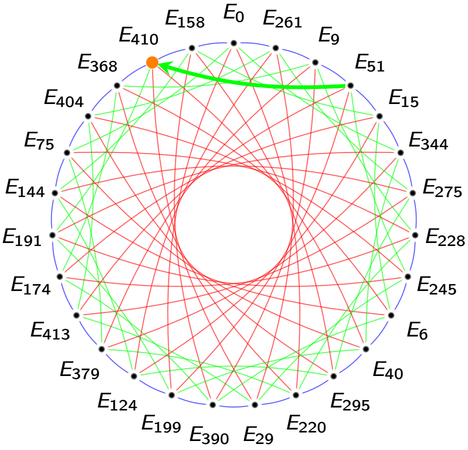
$\uparrow$



Bob

$b = [+, +, -, +]$

$\uparrow$

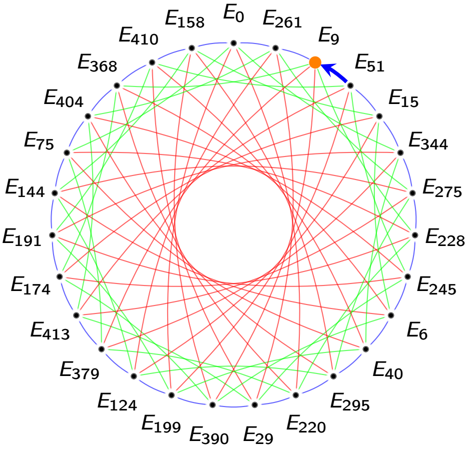


Diffie-Hellman on supersingular isogeny graphs

Alice

$$a = [+, -, +, -]$$

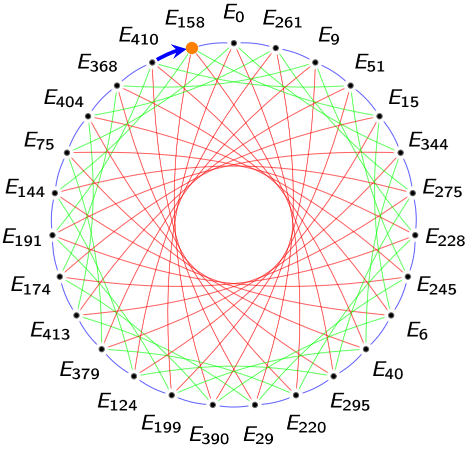
$\uparrow$



Bob

$$b = [+, +, -, +]$$

$\uparrow$



Diffie-Hellman on supersingular isogeny graphs

Alice

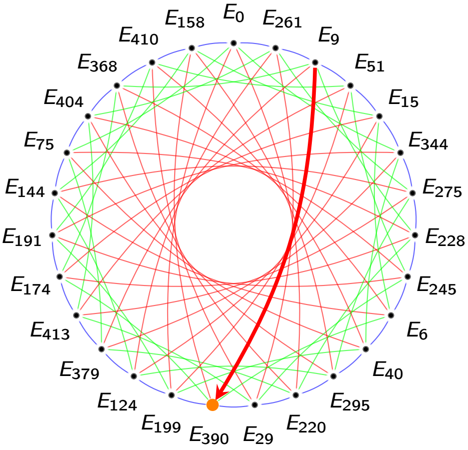
$$a = [+, -, +, -]$$

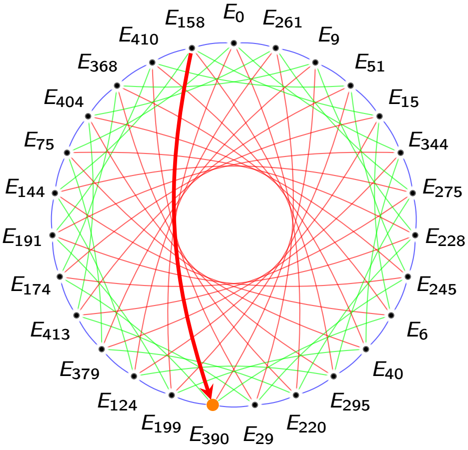
$\uparrow$

Bob

$$b = [+, +, -, +]$$

$\uparrow$





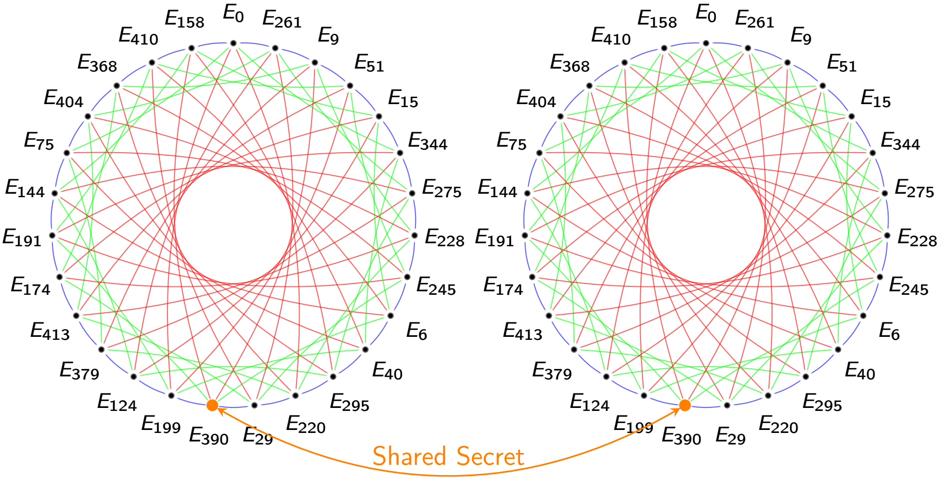
Diffie-Hellman on supersingular isogeny graphs

Alice

$a = [+, -, +, -]$

Bob

$b = [+, +, -, +]$



Practicalities

- Seminar will take place in block format between April 11 - 14th 2022 in SR3.

- Seminar will take place in block format between April 11 - 14$^{\text{th}}$ 2022 in SR3.
- Students will meet one or both instructors in the week March 28 - April 1$^{\text{th}}$ 2022 for discussion and clearing of any doubts.

- Seminar will take place in block format between April 11 - 14th 2022 in SR3.
- Students will meet one or both instructors in the week March 28 - April 1th 2022 for discussion and clearing of any doubts.
- The talk handout is to be submitted in the week April 4 - 8th 2022.

- Seminar will take place in block format between April 11 - 14th 2022 in SR3.
- Students will meet one or both instructors in the week March 28 - April 1th 2022 for discussion and clearing of any doubts.
- The talk handout is to be submitted in the week April 4 - 8th 2022.
- Talk and handout to be delivered in English, on either blackboard or computer presentation.

- Seminar will take place in block format between April 11 - 14^{th} 2022 in SR3.
- Students will meet one or both instructors in the week March 28 - April 1^{th} 2022 for discussion and clearing of any doubts.
- The talk handout is to be submitted in the week April 4 - 8^{th} 2022.
- Talk and handout to be delivered in English, on either blackboard or computer presentation.
- Students can ask questions to the instructors before March 28th also – Proseminar talks to C. V. Sriram, and (Pro)Seminar talks to B. S. Banwait.

Questions?

Fragen?

Intentional bilingual pause slide for questions

Absichtliche zweisprachige Pausenfolie für Fragen