Cryptography

Lecture 15: Final Remarks

February 4, 2025

Overview

Contents

1. Classical cryptography
   (Shift & Vigenère cipher, one-time pad, perfect secrecy)
2. Security definitions & threat models
   (Computational security, CPA & CCA)
3. Private-key cryptography
   (Message authentication, hash functions, primitives, relevant ciphers)
4. Public-key cryptography
   (Assumptions, key management, digital signatures, relevant ciphers)

Interactive Proof Systems

Motivation

- Model communication with potentially unreliable partner
  (adversary or even cheating partner in zero-knowledge proofs)
- Classic model in cryptography
- Deal with adversaries

Interactive Proof Systems

§13.1 Definition (Interactive proof system)

Interactive proof system is pair (A, B) with $C = \{0, 1\}$ and

- arbitrary function $A\colon \bigcup_{i \in \mathbb{N}} (C^*)^{1+2i} \to C^*$ (Alice)
- randomized TM B that is time-bounded by polynomial p (Bob)

Protocol runs for $q(|w|)$ rounds on input w for some polynomial q.
Alice and Bob communicate via common work tape W

- Round i starts with message $a_i = A(w, a_1, b_1, \ldots, a_{i-1}, b_{i-1})$ on W sent by Alice to Bob
- Then Bob replies with message b_i to Alice on W

$$b_i = B(w, a_1, b_1, \ldots, a_{i-1}, b_{i-1}, a_i, Z_1, \ldots, Z_i) \ ,$$

  where Z_i is random sequence of round i
- In last round $q(|w|)$ Bob decides whether to accept w

Notes

- Messages b_i of Bob have polynomial length
  because Bob runs in polynomial time
- Messages a_i of Alice can have unbounded length,
  but Bob B can only read polynomially sized prefix
- Wlog. suppose that $|a_i| \leq p(|w|)$ and $|b_i| \leq p(|w|)$
- Also suppose that Bob does not share his random bits with Alice
  (although this has no essential influence)
- In round i Bob has access to random bits $Z_1, \ldots, Z_i$

Interactive Proof Systems

§13.2 Definition

Let (A, B) be interactive proof system and $L \subseteq \Sigma^*$. Then (A, B) accepts language L if for all $w \in \Sigma^*$

- If $w \in L$, then Bob accepts input w with probability at least $1 - 2^{-|w|}$
- If $w \notin L$, then there exists no interactive proof system (A', B), in which Bob accepts w with probability at least $2^{-|w|}$

$$\mathbf{IP} = \{L \subseteq \Sigma^* \mid \exists \text{ interactive proof system that accepts } L\}$$

Notes

- Introduced by Goldwasser, Micali, and Rackoff in 1985
- Alice is computationally unbounded prover
- Bob is polynomial-time verifier to be convinced

Interactive Proof Systems

Intuition

- Completeness If $w \in L$, then prover that follows protocol convinces verifier almost certainly
- Correctness If $w \notin L$, then no prover, whether it follows protocol or not, will convince verifier with non-negligible probability

§13.3 Theorem

NP $\subseteq$ **IP**

Proof

Let $L \in$ **NP**. For $w \in L$ Alice shares certificate of w with Bob, who verifies validity and accepts w accordingly. Hence Bob only accepts inputs of L and no prover can convince Bob to accept input not belonging to L $\square$

Interactive Proof Systems

Oded Goldreich (* 1957)

- Israelian computer scientist
- Professor of cryptography at Weizmann institute
- Knuth prize in 2017

Silvio Micali (* 1954)

- Italian mathematician & computer scientist
- Professor at MIT
- Gödel prize in 1993

Avi Wigderson (* 1956)

- Israelian mathematician
- Professor at Princeton University
- Gödel prize in 2009

© Ednawig

Graph Isomorphism

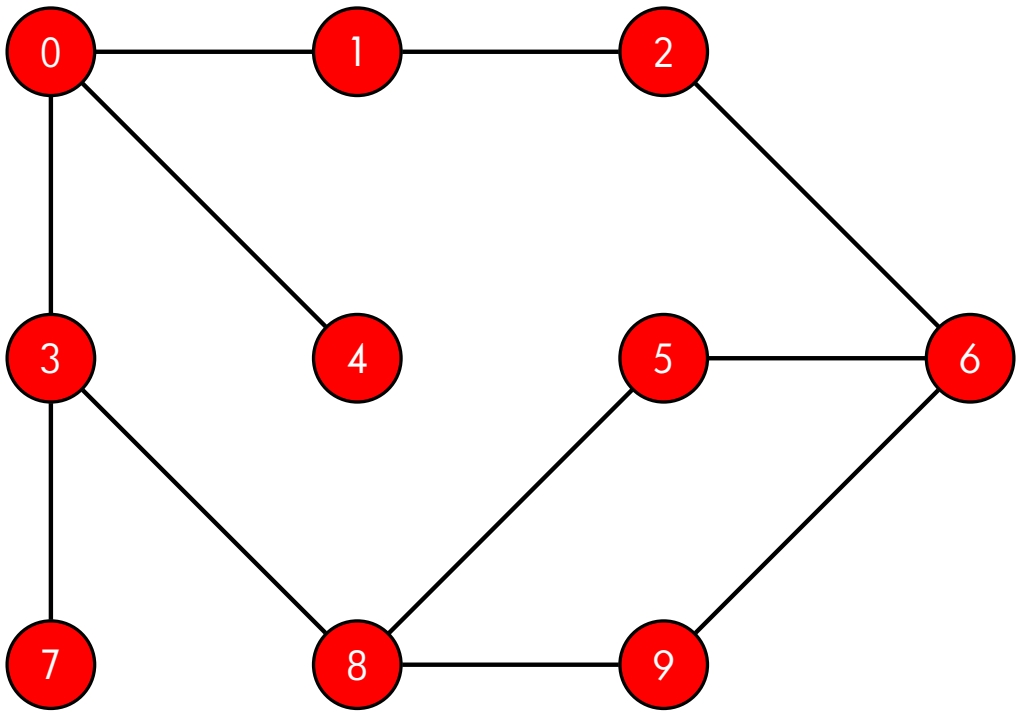
Recall

- Undirected graph is (V, E) with finite set V of vertices and symmetric relation $E \subseteq V \times V$ of edges
- Graphs (V, E) and (V', E') isomorphic if there exists bijection $h\colon V \to V'$ such that

$$(v, v') \in E \qquad \text{if and only if} \qquad \big(h(v), h(v')\big) \in E'$$

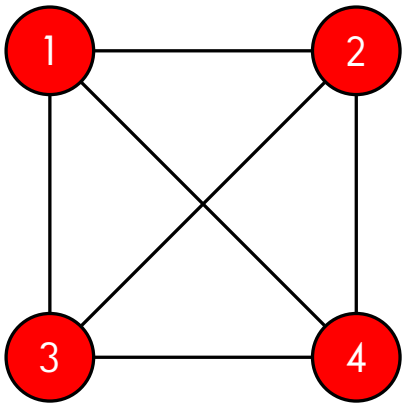
Graph Isomorphism

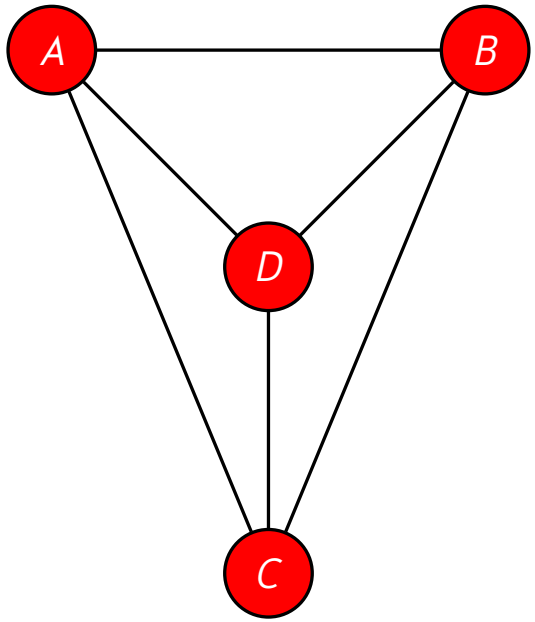
Undirected graph



Graph Isomorphism

Isomorphic?

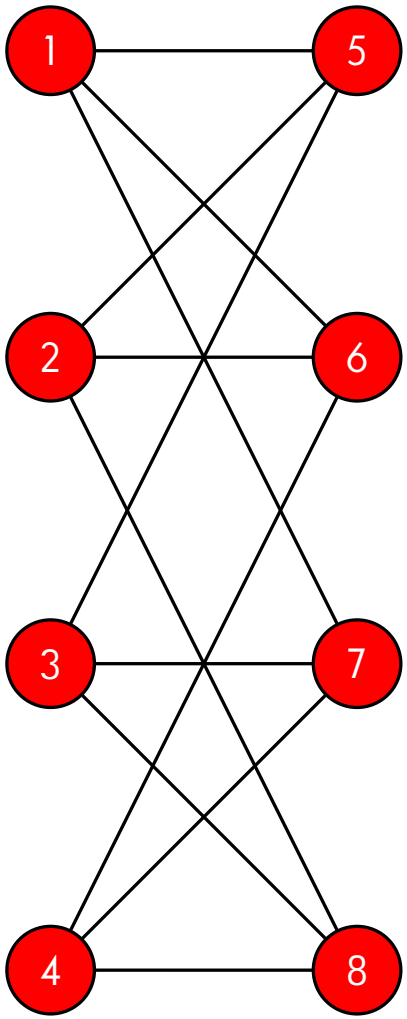


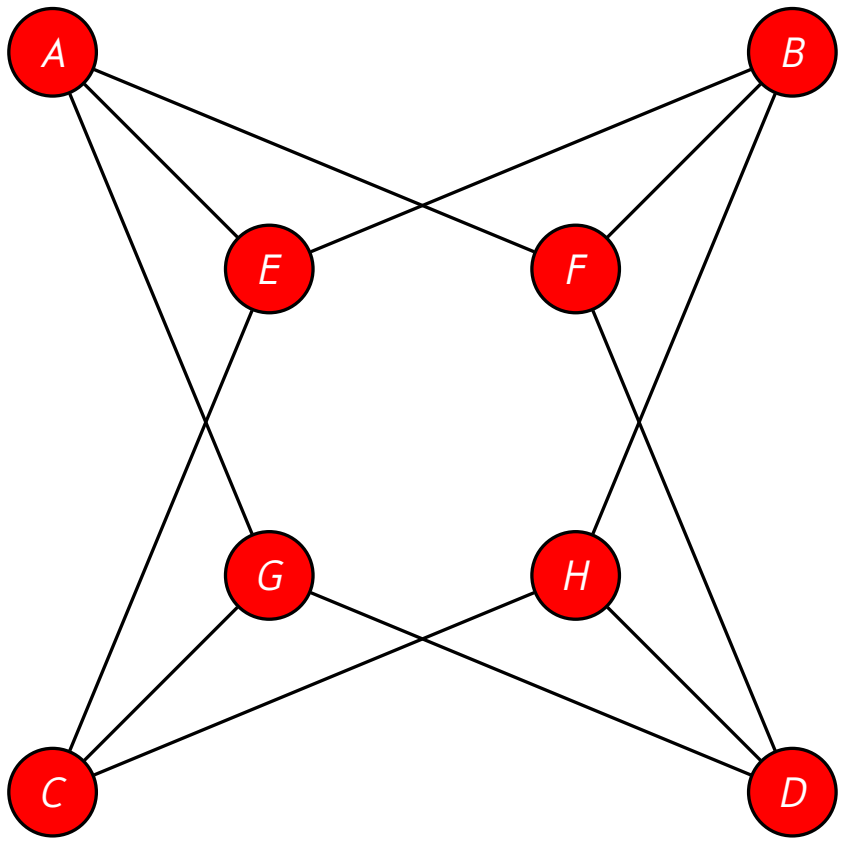


Solution: yes, isomorphism maps $1 \mapsto A$, $2 \mapsto B$, $3 \mapsto C$, and $4 \mapsto D$

Graph Isomorphism

Isomorphic?





Solution: yes, $1 \mapsto E$, $2 \mapsto F$, $3 \mapsto G$, $4 \mapsto H$, $5 \mapsto A$, $6 \mapsto B$, $7 \mapsto C$, $8 \mapsto D$

§13.4 Theorem (Goldreich, Micali, Wigderson 1987)

Non-isomorphism of finite graphs is in **IP**

Proof (1/2)

Let $G_0 = (V, E)$ and $G_1 = (V', E')$ be input graphs known to Alice and Bob. Wlog. let $|E| = |E'|$ because otherwise Bob accepts immediately (since graphs not isomorphic). We additionally assume $V = V'$ and $m = |G_0| + |G_1|$ is size of input. Protocol runs as follows:

1. Bob selects m permutations $\pi_i \in \mathrm{Perm}(\{1, \ldots, |V|\})$ and m bits $b_i \in \{0, 1\}$ randomly for all $1 \leq i \leq m$ and sends $(\pi_1(G_{b_1}), \ldots, \pi_m(G_{b_m}))$
2. Bob expects bit string $c_1, \ldots, c_m \in \{0, 1\}$ as reply and accepts if and only if $b_i = c_i$ for all $1 \leq i \leq m$

Proof (2/2)

- Assume non-isomorphic G_0 and G_1. Alice checks whether permutation $\pi_i(G_{b_i})$ is isomorphic to either G_0 or G_1 and obtains b_i. This yields $b_1, \ldots, b_m$ and Bob accepts with probability 1
- Assume isomorphic G_0 and G_1. Alice receives only permutations of isomorphic graphs G_0 and G_1 and can thus only guess m bits $c_1, \ldots, c_m$. Bob will only accept if $(b_1, \ldots, b_m) = (c_1, \ldots, c_m)$. The probability for this case is 2^{-m}, so he will accept isomorphic graphs only with probability 2^{-m} □

Interactive Proof Systems

Notes

- Isomorphism of finite graphs is in **NP**,
  (guess isomorphism & check validity)
  but membership in **P** as well as **NP**-completeness are unknown
  (expected to be not **NP**-hard)
- Whether non-isomorphism of finite graphs is in **NP** also unknown

- If G_0 and G_1 are isomorphic, then Alice shares no isomorphism with Bob and he also receives no information about such isomorphism
  $\rightarrow$ Zero-knowledge proof

Computing Inverses

Motivation

- Modular arithmetic relevant
  (e.g. calculations with times, days, dates)
- Field $(\mathbb{Z}_p, +, \cdot, 0, 1)$ for prime p (or its multiplicative group)
  used in RSA and even ElGamal
- Insights into integers (number theory)

Computing Inverses — Divisibility

§13.5 Definition (divisor)

Let $a \in \mathbb{N}$ and $b \in \mathbb{Z}$. Then a divides b if there exists $k \in \mathbb{Z}$ with $b = k \cdot a$
We write $a \mid b$ if a divides b

Examples

- $11 \mid 121$ because $121 = 11 \cdot 11$
- $n \mid 0$ for every $n \in \mathbb{N}$ because $0 = 0 \cdot n$
- $2 \nmid 121$ because $\frac{121}{2} = 60.5 \notin \mathbb{Z}$

Computing Inverses — Divisibility

§13.6 Definition (set of divisors)

For every $b \in \mathbb{Z}$ let

$$D_b = \{a \in \mathbb{N} \mid a \mid b\}$$

be set of all nonnegative integers that divide b

Examples

- $D_8 = \{1, 2, 4, 8\}$
- $D_9 = \{1, 3, 9\}$
- $D_{12} = \{1, 2, 3, 4, 6, 12\}$

Computing Inverses — Divisibility

§13.7 Theorem

Let $m \mid b$. Then for all $c \in \mathbb{Z}$ we have $m \mid c$ if and only if $m \mid (b + c)$

Proof (both-sided implications)

($\rightarrow$) Let $m \mid b$ and $m \mid c$. There exists $k, n \in \mathbb{Z}$ such that $b = k \cdot m$ and $c = n \cdot m$. Thus $b + c = km + nm = (k + n) \cdot m$ and hence $m \mid (b + c)$

($\leftarrow$) Let $m \mid b$ and $m \mid (b + c)$. There exists $k, n \in \mathbb{Z}$ such that $b = k \cdot m$ and $b + c = n \cdot m$. Thus $c = (b + c) - b = nm - km = (n - k) \cdot m$ and hence $m \mid c$ $\square$

Computing Inverses — Divisibility

§13.8 Corollary

Let $a, b \in \mathbb{N}$. Then

$$D_a \cap D_b = D_{(a+b)} \cap D_b$$

Proof (both-sided subsets)

($\subseteq$) Let $m \in D_a \cap D_b$. Hence $m \mid a$ and $m \mid b$. By Theorem §13.7 we have $m \mid (a + b)$ and hence $m \in D_{(a+b)} \cap D_b$

($\supseteq$) Let $m \in D_{(a+b)} \cap D_b$. Hence $m \mid (a + b)$ and $m \mid b$. By Theorem §13.7 we have $m \mid a$ and hence $m \in D_a \cap D_b$ □

Notes

- a and b have same divisors as $a + b$ and b
- Also true for divisors of a and b
  and divisors of $a + kb$ (for $k \in \mathbb{N}$) and b

Computing Inverses — Divisibility

§13.9 Theorem

Let $a, b \in \mathbb{N}$. For all $k \in \mathbb{N}$

$$D_a \cap D_b = D_{(a+kb)} \cap D_b$$

Proof (induction on k)

- Base: Trivial for $k = 0$
- Hypothesis: Statement true for k
- Step: By hypothesis we have $D_a \cap D_b = D_{(a+kb)} \cap D_b$. Additionally by Corollary §13.8

$$D_{(a+kb)} \cap D_b = \underbrace{D_{(a+kb+b)}}_{=D_{(a+(k+1)b)}} \cap D_b$$

which proves $D_a \cap D_b = D_{(a+(k+1)b)} \cap D_b$ □

Computing Inverses — Divisibility

Notes

- $D_a \cap D_b \neq \emptyset$ for all $a, b \in \mathbb{N}$ because $1 \in D_a$ and $1 \in D_b$
- $D_a \cap D_b$ finite for all $a, b \in \mathbb{N} \setminus \{0\}$
  because $m \leq a$ and $m \leq b$ for all $m \in D_a \cap D_b$

§13.10 Definition (greatest common divisor)

Let $a, b \in \mathbb{N} \setminus \{0\}$. Then

$$\gcd(a, b) = \max(D_a \cap D_b)$$

is greatest common divisor of a and b
Numbers a and b are co-prime if $\gcd(a, b) = 1$

Computing Inverses — Divisibility

Examples

- $D_8 = \{1, 2, 4, 8\}$ and $D_9 = \{1, 3, 9\}$ and $D_{12} = \{1, 2, 3, 4, 6, 12\}$
- Thus $\gcd(8, 12) = 4$ and $\gcd(8, 9) = 1$
- 1 is co-prime to each positive integer

Computing Inverses — Divisibility

§13.11 Theorem

Let $a, b \in \mathbb{N}$ with $b \geq 1$. There exist unique $k, r \in \mathbb{N}$ such that

$$a = kb + r \qquad \text{and} \qquad 0 \leq r < b$$

Proof (induction on a; 1/2)

We start with existence

- Base: Let $a = 0$. We set $k = 0$ and $r = 0$. Then $a = kb + r$ and $0 \leq r < b$ as desired
- Hypothesis: Let $k, r \in \mathbb{N}$ be such that $a = kb + r$ and $0 \leq r < b$
- Step: We distinguish 2 cases

Proof (induction on a; 2/2)

- Case 1: Let $r+1=b$. Then

$$a+1=(kb+r)+1=kb+b=(k+1)b+0$$

  We let $k'=(k+1)$ and $r'=0$, which yields $a+1=k'b+r'$
- Case 2: Let $r+1<b$. Then $a+1=(kb+r)+1$. We set $k'=k$ and $r'=r+1$, which yields $a+1=k'b+r'$

This proves existence. Let $k,k',r,r'\in\mathbb{N}$ such that $a=kb+r$ and $a=k'b+r'$ as well as $0\leq r<b$ and $0\leq r'<b$. Thus $kb+r=k'b+r'$ and thus $(k-k')b=r'-r$. We again distinguish 2 cases

- Let $k-k'=0$. Then $k=k'$ and thus $0=r'-r$, which proves $r=r'$
- Let $k-k'\neq 0$. Then $|(k-k')b|\geq b$, but $|r'-r|<b$. Thus this case is contradictory because $(k-k')b=r'-r$ is impossible $\square$

Computing Inverses — Divisibility

§13.12 Definition (modulo operation)

Let $a, b \in \mathbb{N}$ with $b \geq 1$. By Theorem §13.11 there exists unique $k, r \in \mathbb{N}$ such that $a = kb + r$ and $0 \leq r < b$.
We write $r = a \bmod b$

Examples

- $5 \bmod 2 = 1$ and $12 \bmod 2 = 0$
- $7 \bmod 4 = 3$ and $9 \bmod 4 = 1$

Computing Inverses — Modular Arithmetic

§13.13 Theorem

Let $a, b \in \mathbb{N}$ and $m \in \mathbb{N}$. The following are equivalent

- $r_a = r_b$ where $r_a = a \bmod m$ and $r_b = b \bmod m$
- $m \mid (a - b)$

Proof (direct)

Since $|r_a - r_b| < m$ we have

$$
\begin{aligned}
& m \mid (a - b) \\
\text{iff } & \exists k \big(k \in \mathbb{Z} \wedge a - b = km \big) \\
\text{iff } & \exists k \Big(k \in \mathbb{Z} \wedge \big(\lfloor \tfrac{a}{m} \rfloor \cdot m + r_a \big) - \big(\lfloor \tfrac{b}{m} \rfloor \cdot m + r_b \big) = km \Big) \\
\text{iff } & \exists k \Big(k \in \mathbb{Z} \wedge r_a - r_b = (k - \lfloor \tfrac{a}{m} \rfloor + \lfloor \tfrac{b}{m} \rfloor) \cdot m \Big) \\
\text{iff } & r_a = r_b
\end{aligned}
$$

□

Computing Inverses — Modular Arithmetic

§13.14 Theorem

Let $a, b, c, d \in \mathbb{N}$ and $m \in \mathbb{N} \setminus \{0\}$ such that $a \bmod m = b \bmod m$ and $c \bmod m = d \bmod m$. Then

1. $(a + c) \bmod m = (b + d) \bmod m$
2. $(a \cdot c) \bmod m = (b \cdot d) \bmod m$

Proof (direct)

Easy exercise □

Computing Inverses — Modular Arithmetic

Notes

- Basic rule for modular arithmetic
- Always compute with "small numbers" (from $\{0, 1, \ldots, m-1\}$)

$\rightarrow$ Modular arithmetic simpler than traditional arithmetic in $\mathbb{N}$

Computing Inverses — Modular Arithmetic

§13.15 Theorem

Let $a, b, c \in \mathbb{N}$ and $m \in \mathbb{N} \setminus \{0\}$ such that c and m are co-prime.
If $(a \cdot c) \bmod m = (b \cdot c) \bmod m$, then $a \bmod m = b \bmod m$

Proof (direct)

By Theorem §13.13 we have $m \mid (ac - bc)$ and thus $m \mid (a - b)c$.
Since m and c are co-prime we must have $m \mid (a - b)$ (follows from prime factorization). By Theorem §13.13 we thus have $a \bmod m = b \bmod m$ □

Computing Inverses — Euclidian Algorithm

Motivation

- Algorithm for computation of greatest common divisor
- Very efficient
- Ancient; Euclid presents method (unclear whether he discovered it)
- Oldest non-trivial algorithm

Euclid of Alexandria (3rd century B.C.)

- Greek mathematician
- Collected knowledge of mathematics
- Promoter of strict proofs

Computing Inverses — Euclidian Algorithm

§13.16 Theorem

For every $a, b \in \mathbb{N} \setminus \{0\}$

$$\gcd(a, b) = \gcd(a \bmod b, b)$$

Proof (direct)

By Theorem §13.11 there exists unique $k \in \mathbb{N}$ such that $a = kb + r$ with $r = a \bmod b$. By Theorem §13.9

$$D_{(a \bmod b)} \cap D_b = \underbrace{D_{((a \bmod b)+kb)}}_{D_a} \cap D_b = D_a \cap D_b \ ,$$

which also proves that their maximal elements coincide □

Computing Inverses — Euclidian Algorithm

§13.17 Definition (recursive computation of gcd)

Define mapping $e\colon \mathbb{N}_+ \times \mathbb{N}_+ \to \mathbb{N}_+$ recursively for all $a, b \in \mathbb{N}_+$ by

$$e(a,b) = \begin{cases} b & \text{if } a \bmod b = 0 \\ e(b, a \bmod b) & \text{otherwise} \end{cases}$$

a	b	$a \bmod b$
127	34	25
34	25	9
25	9	7
9	7	2
7	2	1
2	1	0

- We compute $e(127, 34)$
- Since $127 \bmod 34 = 25 \to e(34, 25)$
- Since $34 \bmod 25 = 9 \to e(25, 9)$
- Since $25 \bmod 9 = 7 \to e(9, 7)$
- Since $9 \bmod 7 = 2 \to e(7, 2)$
- Since $7 \bmod 2 = 1 \to e(2, 1)$
- Since $2 \bmod 1 = 0$ we return 1

Computing Inverses — Euclidian Algorithm

§13.18 Theorem

For all $a, b \in \mathbb{N}_+$ we have $e(a, b) = \gcd(a, b)$

Proof (induction on b; 1/2)

- Base: Let $b = 1$. Then $\gcd(a, b) = b$ and thus $e(a, b) = b$ because $a \bmod b = a \bmod 1 = 0$
- Hypothesis: Statement true for b and all smaller values
- Step: We distinguish several cases
  - Let $a \bmod (b+1) = 0$. Then $e(a, b+1) = b+1$ and thus $\gcd(a, b+1) = b+1$ since $b+1$ is divisor of a and obviously greatest divisor of $b+1$
  - Let $a \bmod (b+1) \neq 0$. We distinguish two additional cases

Computing Inverses — Euclidian Algorithm

Proof (induction; 2/2)

We are in induction step and have $a \bmod (b+1) \neq 0$

- Suppose $a < b+1$. Then $a \bmod (b+1) = a$ and thus $e(a, b+1) = e(b+1, a)$. Since $a < b+1$ we have $a \leq b$ and from hypothesis we get $e(b+1, a) = \gcd(b+1, a)$. Hence

$$e(a, b+1) = e(b+1, a) = \gcd(b+1, a) = \gcd(a, b+1)$$

- Finally, suppose that $a > b+1$. Then $a \bmod (b+1) < b+1$ and thus

$$\begin{aligned} e(a, b+1) &= e(b+1, a \bmod (b+1)) && \text{(Def. } e\text{)} \\ &= \gcd(b+1, a \bmod (b+1)) && \text{(hypothesis)} \\ &= \gcd(a \bmod (b+1), b+1) \\ &= \gcd(a, b+1) && \text{(Thm. §13.16)} \end{aligned}$$

which completes the induction □

Computing Inverses — Euclidian Algorithm

Computation of $e(16,607,184,\ 2,367,488)$

a	b	$a \bmod b$
16,607,184	2,367,488	34,768
2,367,488	34,768	3,264
34,768	3,264	2,128
3,264	2,128	1,136
2,128	1,136	992
1,136	992	144
992	144	128
144	128	16
128	16	0

Computing Inverses — Euclidian Algorithm

§13.19 Theorem

For all $a, b \in \mathbb{N}_+$ there exist $m, n \in \mathbb{Z}$ such that $e(a, b) = ma + nb$

Proof (induction; 1/2)

- Base: Computation of $e(a, b)$ terminates immediately. Then $a \bmod b = 0$ and $e(a, b) = b$. We set $m = 0$ and $n = 1$. Then $e(a, b) = b = ma + nb$
- Hypothesis: Statement true for calls of e that terminate in k steps
- Step: Let $e(a, b)$ be call that terminates in $k + 1$ steps. Obviously $e(a, b) = e(b, a \bmod b)$ and by hypothesis there exist $m, n \in \mathbb{Z}$ such that $e(b, a \bmod b) = mb + n(a \bmod b)$

$$\begin{aligned} mb + n(a \bmod b) &= mb + n(a - \lfloor \tfrac{a}{b} \rfloor \cdot b) \\ &= na + (m - n \cdot \lfloor \tfrac{a}{b} \rfloor) \cdot b \end{aligned}$$

Computing Inverses — Euclidian Algorithm

Proof (induction; 2/2)

Thus

$$\begin{aligned} e(a,b) &= \gcd(a,b) && \text{(Thm. §13.18)} \\ &= \gcd(a \bmod b, b) && \text{(Thm. §13.16)} \\ &= \gcd(b, a \bmod b) \\ &= e(b, a \bmod b) && \text{(Thm. §13.18)} \\ &= mb + n(a \bmod b) \\ &= na + (m - n \cdot \lfloor \tfrac{a}{b} \rfloor) \cdot b \end{aligned}$$

which proves statement for $m' = n$ and $n' = m - n\lfloor \frac{a}{b} \rfloor$ □

Uniqueness of m and n only modulo a and b

$$e(a,b) = ma + nb = ma + nb + \underbrace{ab - ab}_{=0} = (m+b)a + (n-a)b$$

Computing Inverses — Extended Euclidian Algorithm

§13.20 Definition (extended Euclidian algorithm)

Define mapping $f : \mathbb{N}_+ \times \mathbb{N}_+ \to \mathbb{N}_+ \times \mathbb{Z} \times \mathbb{Z}$ inductively for all $a, b \in \mathbb{N}_+$ by

$$f(a,b) = \begin{cases} (b, 0, 1) & \text{if } a \bmod b = 0 \\ (d, n, m - n\lfloor \frac{a}{b} \rfloor) & \text{otherwise,} \\ & \text{where } (d, m, n) = f(b, a \bmod b) \end{cases}$$

Notes

- Correctness results directly from Theorem §13.19
- (Relevant part) discovered by Claude Bachet
- Extension to polynomials by Étienne Bézout

Computing Inverses — Extended Euclidian Algorithm

Claude Gaspard Bachet (* 1581; † 1638)

- French mathematician
- Worked on number theory
- Wrote book for Fermat's notes

Étienne Bézout (* 1730; † 1783)

- French mathematician
- Inspired by Leonhard Euler
- Worked for military

Computing Inverses — Extended Euclidian Algorithm

Computation of $f(16,607,184,\ 2,367,488)$

a	b	$a \bmod b$	d	m	n	$m - n\lfloor \frac{a}{b} \rfloor$
16,607,184	2,367,488	34,768				
2,367,488	34,768	3,264				
34,768	3,264	2,128				
3,264	2,128	1,136				
2,128	1,136	992				
1,136	992	144				
992	144	128				
144	128	16	16	0	1	−1
128	16	0				

We get $f(16,607,184,\ 2,367,488)$

Computing Inverses — Extended Euclidian Algorithm

Computation of $f(16,607,184,\ 2,367,488)$

a	b	$a \bmod b$	d	m	n	$m - n\lfloor \frac{a}{b} \rfloor$
16,607,184	2,367,488	34,768				
2,367,488	34,768	3,264				
34,768	3,264	2,128				
3,264	2,128	1,136				
2,128	1,136	992				
1,136	992	144				
992	144	128	16	1	−1	7
144	128	16	16	0	1	−1
128	16	0				

We get $f(16,607,184,\ 2,367,488)$

Computing Inverses — Extended Euclidian Algorithm

Computation of $f(16,607,184,\ 2,367,488)$

a	b	$a \bmod b$	d	m	n	$m - n\lfloor \frac{a}{b} \rfloor$
16,607,184	2,367,488	34,768				
2,367,488	34,768	3,264				
34,768	3,264	2,128				
3,264	2,128	1,136				
2,128	1,136	992				
1,136	992	144	16	−1	7	−8
992	144	128	16	1	−1	7
144	128	16	16	0	1	−1
128	16	0				

We get $f(16,607,184,\ 2,367,488)$

Computing Inverses — Extended Euclidian Algorithm

Computation of $f(16,607,184,\ 2,367,488)$

a	b	$a \bmod b$	d	m	n	$m - n\lfloor \frac{a}{b} \rfloor$
16,607,184	2,367,488	34,768				
2,367,488	34,768	3,264				
34,768	3,264	2,128				
3,264	2,128	1,136				
2,128	1,136	992	16	7	−8	15
1,136	992	144	16	−1	7	−8
992	144	128	16	1	−1	7
144	128	16	16	0	1	−1
128	16	0				

We get $f(16,607,184,\ 2,367,488)$

Computing Inverses — Extended Euclidian Algorithm

Computation of $f(16,607,184,\ 2,367,488)$

a	b	$a \bmod b$	d	m	n	$m - n\lfloor \frac{a}{b} \rfloor$
16,607,184	2,367,488	34,768				
2,367,488	34,768	3,264				
34,768	3,264	2,128				
3,264	2,128	1,136	16	−8	15	−23
2,128	1,136	992	16	7	−8	15
1,136	992	144	16	−1	7	−8
992	144	128	16	1	−1	7
144	128	16	16	0	1	−1
128	16	0				

We get $f(16,607,184,\ 2,367,488)$

Computing Inverses — Extended Euclidian Algorithm

Computation of $f(16,607,184,\ 2,367,488)$

a	b	$a \bmod b$	d	m	n	$m - n\lfloor \frac{a}{b} \rfloor$
16,607,184	2,367,488	34,768				
2,367,488	34,768	3,264				
34,768	3,264	2,128	16	15	−23	245
3,264	2,128	1,136	16	−8	15	−23
2,128	1,136	992	16	7	−8	15
1,136	992	144	16	−1	7	−8
992	144	128	16	1	−1	7
144	128	16	16	0	1	−1
128	16	0				

We get $f(16,607,184,\ 2,367,488)$

Computing Inverses — Extended Euclidian Algorithm

Computation of $f(16,607,184,\ 2,367,488)$

a	b	$a \bmod b$	d	m	n	$m - n\lfloor \frac{a}{b} \rfloor$
16,607,184	2,367,488	34,768				
2,367,488	34,768	3,264	16	−23	245	−16,683
34,768	3,264	2,128	16	15	−23	245
3,264	2,128	1,136	16	−8	15	−23
2,128	1,136	992	16	7	−8	15
1,136	992	144	16	−1	7	−8
992	144	128	16	1	−1	7
144	128	16	16	0	1	−1
128	16	0				

We get $f(16,607,184,\ 2,367,488)$

Computing Inverses — Extended Euclidian Algorithm

Computation of $f(16,607,184,\ 2,367,488)$

a	b	$a \bmod b$	d	m	n	$m - n\lfloor \frac{a}{b} \rfloor$
16,607,184	2,367,488	34,768	16	245	−16,683	117,026
2,367,488	34,768	3,264	16	−23	245	−16,683
34,768	3,264	2,128	16	15	−23	245
3,264	2,128	1,136	16	−8	15	−23
2,128	1,136	992	16	7	−8	15
1,136	992	144	16	−1	7	−8
992	144	128	16	1	−1	7
144	128	16	16	0	1	−1
128	16	0				

We get $f(16,607,184,\ 2,367,488)$

Computing Inverses — Extended Euclidian Algorithm

Computation of $f(16,607,184,\ 2,367,488)$

a	b	$a \bmod b$	d	m	n	$m - n\lfloor \frac{a}{b} \rfloor$
16,607,184	2,367,488	34,768	16	245	−16,683	117,026
2,367,488	34,768	3,264	16	−23	245	−16,683
34,768	3,264	2,128	16	15	−23	245
3,264	2,128	1,136	16	−8	15	−23
2,128	1,136	992	16	7	−8	15
1,136	992	144	16	−1	7	−8
992	144	128	16	1	−1	7
144	128	16	16	0	1	−1
128	16	0				

We get $f(16,607,184,\ 2,367,488) = (16,\ -16,683,\ 117,026)$

(Indeed $-16,683 \cdot 16,607,184 + 117,026 \cdot 2,367,488 = 16$)

Computing Inverses

Computing inverses

- Let $\ell \in \mathbb{N}_+$
- Recall $i \in \mathbb{Z}_\ell$ is invertable (i.e., $i \in \mathbb{Z}_\ell^*$) iff $\gcd(i, \ell) = 1$ (Lm. §9.10)

- Let us compute inverse of $i \in \mathbb{Z}_\ell^*$
- Suppose that $f(i, \ell) = (1, m, n)$ (extended Euclidian algorithm)
  (first component is 1 because $\gcd(i, \ell) = 1$)
- Hence $1 = mi + n\ell$ and thus $1 = mi \bmod \ell$ so

$$i^{-1} = m \bmod \ell$$

- Inverse of $i \in \mathbb{Z}_\ell^*$ is $i^{-1} = m \bmod \ell$

Summary

- Graph isomorphism
- Zero-knowledge proofs
- Modular arithmetic
- Euclidian algorithm & extended Euclidian algorithm
- Computing multiplicative inverses in $\mathbb{Z}_\ell$

All the best for exam!