

# CSE 291-E: Applied Cryptography

**Nadia Heninger**

UCSD

Fall 2020 Lecture 8

## Legal Notice

The Zoom session for this class will be recorded and made available asynchronously on Canvas to registered students.

# Announcements

1. HW 3 is due today!
2. HW 4 is due before class in 1 week, October 3.

**Last time:** Authenticated encryption

**This time:** Number theory review

# Fundamental theorem of arithmetic

## Theorem

*Every  $n \in \mathbb{Z}$   $n \neq 0$  has unique factorization  $n = \pm p_1^{e_1} p_2^{e_2} \dots p_r^{e_r}$  with  $p_i$  distinct primes and  $e_i$  positive integers.*

## Division and remainder

### Theorem

$a, b \in \mathbb{Z}$ ,  $b > 0$ ,  $\exists$  unique  $q, r \in \mathbb{Z}$  s.t.  $a = bq + r$ ,  $0 \leq r < b$ .

$$r \equiv a \pmod{b} \quad a \pmod{b} = a - b \lfloor \frac{a}{b} \rfloor$$

Because we're in CS, we also write  $r = a \pmod{b}$ .

$$b \mid a \iff a \pmod{b} = 0$$

$$a = b \pmod{N}: (a \pmod{N}) = (b \pmod{N})$$

$$a = b \pmod{N} \iff N \mid (a - b)$$

## GCDs and Extended Euclidean Algorithm

$\gcd(a, b)$  : greatest common divisor  $d$  s.t.  $d \mid a$  and  $d \mid b$

**Theorem (Extended Euclidean Algorithm)**

$a, b \in \mathbb{Z}$  (and positive)  $\exists x, y \in \mathbb{Z}$  s.t.  $ax + by = \gcd(a, b)$

# GCDs and Extended Euclidean Algorithm

$\gcd(a, b)$  : greatest common divisor  $d$  s.t.  $d \mid a$  and  $d \mid b$

## Theorem (Extended Euclidean Algorithm)

$a, b \in \mathbb{Z}$  (and positive)  $\exists x, y \in \mathbb{Z}$  s.t.  $ax + by = \gcd(a, b)$

### Proof.

Let  $I = \{sa + rb \mid r, s \in \mathbb{Z}\}$  Let  $d$  be the smallest positive elt. of  $I$ .

- $d$  divides every element of  $I$ :

1. Choose  $c = s_c a + r_c b$ .

2.  $c = qd + r$ :

$$r = c - qd = s_c a + r_c b - q(ax + by) = (s_c - qx)a + (r_c - qy)b \in I$$

Thus  $r = 0$  by minimality of  $d$ , thus  $d \mid c$ .



# GCDs and Extended Euclidean Algorithm

$\gcd(a, b)$  : greatest common divisor  $d$  s.t.  $d \mid a$  and  $d \mid b$

## Theorem (Extended Euclidean Algorithm)

$a, b \in \mathbb{Z}$  (and positive)  $\exists x, y \in \mathbb{Z}$  s.t.  $ax + by = \gcd(a, b)$

### Proof.

Let  $I = \{sa + rb \mid r, s \in \mathbb{Z}\}$  Let  $d$  be the smallest positive elt. of  $I$ .

- $d$  divides every element of  $I$ :

1. Choose  $c = s_c a + r_c b$ .

2.  $c = qd + r$ :

$$r = c - qd = s_c a + r_c b - q(ax + by) = (s_c - qx)a + (r_c - qy)b \in I$$

Thus  $r = 0$  by minimality of  $d$ , thus  $d \mid c$ .

- $d$  is largest: Assume  $\exists d' > d$  s.t.  $d' \mid a$ ,  
 $d' \mid b \implies d' \mid xa + yb$   
 $\implies d' \mid d$  but  $d' > d$  contradiction.



## Math version: Ideals

We defined  $I = \{sa + rb \mid r, s \in \mathbb{Z}\}$ .

$I$  is an *ideal* of  $\mathbb{Z}$ .

- $I \subseteq \mathbb{Z}$
- Closed under addition:  $c, d \in I \implies c + d \in I$
- Closed under multiplication in  $\mathbb{Z}$ :  $c \in I, z \in \mathbb{Z} \implies cz \in I$

## Math version: Ideals

We defined  $I = \{sa + rb \mid r, s \in \mathbb{Z}\}$ .

$I$  is an *ideal* of  $\mathbb{Z}$ .

- $I \subseteq \mathbb{Z}$
- Closed under addition:  $c, d \in I \implies c + d \in I$
- Closed under multiplication in  $\mathbb{Z}$ :  $c \in I, z \in \mathbb{Z} \implies cz \in I$

Facts about ideals:

- $0 \in I$ :  $0 \cdot c = 0 \in I$
- $c \in I \implies -c \in I$ :  $c \cdot (-1) = -c \in I$
- $c, d \in I \implies c - d \in I$

# Principal ideals

More facts about ideals:

- $\{0\}$  is an ideal

# Principal ideals

More facts about ideals:

- $\{0\}$  is an ideal
- $\mathbb{Z}$  is an ideal

# Principal ideals

More facts about ideals:

- $\{0\}$  is an ideal
- $\mathbb{Z}$  is an ideal
- $1 \in I \implies I = \mathbb{Z}$

# Principal ideals

More facts about ideals:

- $\{0\}$  is an ideal
- $\mathbb{Z}$  is an ideal
- $1 \in I \implies I = \mathbb{Z}$
- $a\mathbb{Z} = \{az \mid z \in \mathbb{Z}\} = \text{“ideal generated by } a\text{”}$

# Principal ideals

More facts about ideals:

- $\{0\}$  is an ideal
- $\mathbb{Z}$  is an ideal
- $1 \in I \implies I = \mathbb{Z}$
- $a\mathbb{Z} = \{az \mid z \in \mathbb{Z}\} = \text{“ideal generated by } a\text{”}$
- “principal ideal”: ideal of form  $a\mathbb{Z}$

## Theorem

*All ideals of  $\mathbb{Z}$  are principal.*

In short:

$$a\mathbb{Z} + b\mathbb{Z} = d\mathbb{Z}, \quad d = \gcd(a, b)$$



# Euclidean Algorithm

**Input:**  $a, b \in \mathbb{Z}$

**Output:**  $d = \gcd(a, b)$

If  $b = 0$ :

    return  $a$

else:

    return  $\gcd(b, a \bmod b)$

# Extended Euclidean Algorithm

**Input:**  $a, b \in \mathbb{Z}$

**Output:**  $d, x, y$  with  $d = \gcd(a, b)$ ,  $ax + by = d$

If  $b \mid a$ :

return  $b, 0, 1$

else:

compute  $a = qb + r$

$d, x, y = \text{egcd}(b, r)$       ( $xb + yr = d$ )

return  $(d, y, x - yq)$

## Theorem

*The Extended Euclidean Algorithm runs in time  $O(\lg(a) \lg(b))$ .*

## Theorem

*If  $c \mid ab$ ,  $\gcd(a, c) = 1 \implies c \mid b$*

## Modular inverses

Inverse of  $b \bmod N$ :  $bb^{-1} \equiv 1 \bmod N$

- Not defined if  $b$  not invertible.
- 0 has no inverse.

### Theorem

$a$  invertible mod  $N \iff \gcd(a, N) = 1$

## Modular inverses

Inverse of  $b \bmod N$ :  $bb^{-1} \equiv 1 \bmod N$

- Not defined if  $b$  not invertible.
- 0 has no inverse.

### Theorem

$a$  invertible mod  $N \iff \gcd(a, N) = 1$

### Proof.

$\implies$

$$ab \equiv 1 \bmod N$$

$$ab = 1 + cN$$

$$ab - cN = 1 \implies \gcd(a, N) = 1$$

## Modular inverses

Inverse of  $b \bmod N$ :  $bb^{-1} \equiv 1 \bmod N$

- Not defined if  $b$  not invertible.
- 0 has no inverse.

### Theorem

$a$  invertible mod  $N \iff \gcd(a, N) = 1$

### Proof.

$\implies$

$$ab \equiv 1 \bmod N$$

$$ab = 1 + cN$$

$$ab - cN = 1 \implies \gcd(a, N) = 1$$

$$\iff ax + Ny = 1 \implies x = a^{-1} \bmod N$$



Implication: Can compute modular inverses using extended GCD.

# Groups

Group:  $(S, \circ)$  with  $S$  a set and  $\circ$  an operation

$G$  is a group if:

- closed under operation  $\circ$
- identity:  $\exists e \in G$  s.t.  $e \circ g = g = g \circ e \forall g \in G$
- inverses:  $\forall g \in G \quad \exists h \in G$  s.t.  $g \circ h = e = h \circ g$
- associative:  $(g_1 \circ g_2) \circ g_3 = g_1 \circ (g_2 \circ g_3)$

# Groups

Group:  $(S, \circ)$  with  $S$  a set and  $\circ$  an operation

$G$  is a group if:

- closed under operation  $\circ$
- identity:  $\exists e \in G$  s.t.  $e \circ g = g = g \circ e \forall g \in G$
- inverses:  $\forall g \in G \exists h \in G$  s.t.  $g \circ h = e = h \circ g$
- associative:  $(g_1 \circ g_2) \circ g_3 = g_1 \circ (g_2 \circ g_3)$

Abelian group:

- commutative:  $\forall g, h : g \circ h = h \circ g$

# Groups

Group:  $(S, \circ)$  with  $S$  a set and  $\circ$  an operation

$G$  is a group if:

- closed under operation  $\circ$
- identity:  $\exists e \in G$  s.t.  $e \circ g = g = g \circ e \forall g \in G$
- inverses:  $\forall g \in G \quad \exists h \in G$  s.t.  $g \circ h = e = h \circ g$
- associative:  $(g_1 \circ g_2) \circ g_3 = g_1 \circ (g_2 \circ g_3)$

Abelian group:

- commutative:  $\forall g, h : \quad g \circ h = h \circ g$

Cyclic group:

- $G = (\langle a \rangle, \circ)$  (G is generated by one element)



## Examples of groups

- $\mathbb{Z}$  abelian group with  $+$   
identity = 0, inverse =  $-g$ , cyclic, generated by 1

## Examples of groups

- $\mathbb{Z}$  abelian group with  $+$   
identity = 0, inverse =  $-g$ , cyclic, generated by 1
- $\mathbb{Z}$  not a group with  $\times$

## Examples of groups

- $\mathbb{Z}$  abelian group with  $+$   
identity = 0, inverse =  $-g$ , cyclic, generated by 1
- $\mathbb{Z}$  not a group with  $\times$
- $(\mathbb{Z} \bmod N, +)$  is a group  
cyclic, generated by 1

## Examples of groups

- $\mathbb{Z}$  abelian group with  $+$   
identity = 0, inverse =  $-g$ , cyclic, generated by 1
- $\mathbb{Z}$  not a group with  $\times$
- $(\mathbb{Z} \bmod N, +)$  is a group  
cyclic, generated by 1
- $(\mathbb{Z} \bmod N, \times)$  is not a group

## Examples of groups

- $\mathbb{Z}$  abelian group with  $+$   
identity = 0, inverse =  $-g$ , cyclic, generated by 1
- $\mathbb{Z}$  not a group with  $\times$
- $(\mathbb{Z} \bmod N, +)$  is a group  
cyclic, generated by 1
- $(\mathbb{Z} \bmod N, \times)$  is not a group
- $(\{1, 2, \dots, p-1\} \bmod p, \times)$  is a group if  $p$  prime  
“multiplicative group mod  $p$ ”  $\mathbb{Z}_p^*$

## Multiplicative group mod $p$

$\mathbb{Z}_p^* = (\{1, 2, \dots, p-1\} \text{ mod } p, \times)$  is a group if  $p$  is prime

Is cyclic:  $\exists a$  s.t.  $G = \langle a \rangle = \{a, a^2, a^3, \dots, a^{p-1}\}$ .

Not every  $a \in G$  generates  $G$ :  $\langle g \rangle$  might be a subgroup of  $G$ .

## Group orders in $\mathbb{Z}_p^*$

Group orders:

- $|G|$  is called the *order* of the group
- The order of an element  $g$  is  $|\langle g \rangle|$

**Theorem (Lagrange)**

$$\text{order}(g) \mid p - 1$$

Example:

$2^0$	$2^1$	$2^2$	$2^3$	$2^4$	$2^5$	$2^6$	
1	2	4	1	2	4	1	mod 7
<hr/>							
$3^0$	$3^1$	$3^2$	$3^3$	$3^4$	$3^5$	$3^6$	
1	3	2	6	4	5	1	mod 7

$\exists$  efficient p.p.t alg. to find generator if factorization of  $p - 1$  is known

# Fermat's little theorem

## Theorem

$G$  an abelian group with  $|G| = m \implies g^m = 1 \forall g \in G$

## Proof.

$$g_1 \circ g_2 \circ \cdots \circ g_m = (gg_1) \circ (gg_2) \circ \cdots \circ (gg_m) = g^m \circ (g_1 \circ g_2 \cdots g_m)$$

Multiply each side by  $g_1^{-1} \circ g_2^{-1} \cdots g_m^{-1}$ .





# Fermat's little theorem

## Theorem

$G$  an abelian group with  $|G| = m \implies g^m = 1 \forall g \in G$

## Proof.

$$g_1 \circ g_2 \circ \cdots \circ g_m = (gg_1) \circ (gg_2) \circ \cdots \circ (gg_m) = g^m \circ (g_1 \circ g_2 \cdots g_m)$$

Multiply each side by  $g_1^{-1} \circ g_2^{-1} \cdots g_m^{-1}$ .



## Corollary (Fermat's little theorem)

$p$  prime,  $a^{p-1} \equiv 1 \pmod{p}$

# Computations with modular arithmetic

Efficient:

- Addition
- Subtraction
- Multiplication
- Inversion (using GCD algorithm)
- Modular exponentiation

# Efficient modular exponentiation

Inefficient exponentiation:

$$g^a = g \cdot g \dots g \quad a \text{ times: not poly-time in } \lg a$$

Efficient exponentiation: Square and multiply (left-to-right)

**Input:** base  $b$ , exponent  $a$ , modulus  $m$

**Output:**  $b^a \bmod m$

Algorithm:

result = 1

for  $i$  from  $\ell \dots 0$  ( $a$  has  $\ell$  bits)

    result = result<sup>2</sup> mod  $m$

    if  $a[i] = 1$ : (bit  $i$  of  $a$  is 1)

        result = result  $\cdot b$  mod  $m$

return result

# Discrete log

“Inverse operation” for modular exponentiation.

No general-purpose efficient algorithms.

## Discrete log

Given  $y, g$  find  $x$  s.t.  $g^x \equiv y \pmod{p}$

Current discrete log record mod  $p$ : 795 bits.

Current factoring record: 830 bits.

Best algorithm: Number field sieve, subexponential time.