Intro to Cryptography

Prepared by Mark on September 25, 2025

Instructor's Handout

Part 1: The Euclidean Algorithm

Definition 1:
The *greatest common divisor* of a and b is the greatest integer that divides both a and b.
We denote this number with $\gcd(a, b)$. For example, $\gcd(45, 60) = 15$.

Problem 2:
Find $\gcd(20, 14)$ by hand.

Solution

$\gcd(20, 14) = 2$

Theorem 3: The Division Algorithm
Given two integers a, b, we can find two integers q, r, where $0 \leq r < b$ and $a = qb + r$.
In other words, we can divide a by b to get q remainder r.

Theorem 4:
For any integers a, b, c,
$\gcd(ac + b, a) = \gcd(a, b)$

Problem 5: The Euclidean Algorithm
Using the two theorems above, detail an algorithm for finding $\gcd(a, b)$.
Then, compute $\gcd(1610, 207)$ by hand.

Solution

Using Theorem 4 and the division algorithm,

$$
\begin{aligned}
&\gcd(1610, 207) \\
&= \gcd(207, 161) \\
&= \gcd(161, 46) \\
&= \gcd(46, 23) \\
&= \gcd(23, 0) = 23
\end{aligned}
\qquad
\begin{aligned}
1610 &= 207 \times 7 + 161 \\
207 &= 161 \times 1 + 46 \\
161 &= 46 \times 3 + 23 \\
46 &= 23 \times 2 + 0
\end{aligned}
$$

Problem 6:
Using the output of the Euclidean algorithm,

- find a pair (u, v) that satisfies $20u + 14v = \gcd(20, 14)$
- find a pair (u, v) that satisfies $541u + 34v = \gcd(541, 34)$

This is called the *extended Euclidean algorithm*.
Hint: You don't need to fully solve the last part of this question.
Understand how you *would* do it, then move on. Don't spend too much time on arithmetic.

Hint:
After running the Euclidean algorithm, you have a table similar to the one shown below.
You can use a bit of algebra to rearrange these statements to get what you need.

Using the Euclidean Algorithm to find that $\gcd(20, 14) = 2$:

$$20 = 14 \times 1 + 6$$
$$14 = 6 \times 2 + 2$$
$$6 = 2 \times 3 + 0$$

We now want to write the 2 in the last equation in terms of 20 and 14.

Solution

Using the output of the Euclidean Algorithm, we can use substitution and a bit of algebra to solve such problems. Consider the following example:

Euclidean Algorithm:
$20 = 14 \times 1 + 6$
$14 = 6 \times 2 + 2$
$6 = 2 \times 3 + 0$

Rearranged:
$6 = 20 - 14 \times 1$
$2 = 14 - 6 \times 2 = \gcd(20, 14)$

Using the right table, we can replace 6 in $2 = 14 - 6 \times 2$ to get $2 = 14 - (20 - 14) \times 2$, which gives us $2 = \gcd(20, 14) = (3)14 + (-2)20$.

$\gcd(20, 14) = 20(-2) + 14(3)$
$\gcd(541, 34) = 541(11) + 34(-175)$

Solution

This problem is too hard. Break it into many.

Part 2: Modular Arithmetic

Definition 7:
$\mathbb{Z}_n$ is the set of integers mod n. For example, $\mathbb{Z}_5 = \{0, 1, 2, 3, 4\}$.

Multiplication in $\mathbb{Z}_n$ works much like multiplication in $\mathbb{Z}$:
If a, b are elements of $\mathbb{Z}_n$, $a \times b$ is the remainder of $a \times b$ when divided by n.
For example, $2 \times 2 = 4$ and $3 \times 4 = 12 = 2$ in $\mathbb{Z}_5$

Problem 8:
Create a multiplication table for $\mathbb{Z}_4$:

$\times$	0	1	2	3
0	?	?	?	?
1	?	?	?	?
2	?	?	?	?
3	?	?	?	?

Definition 9:
Let a, b be elements of If $a \times b = 1$, we say that b is the *inverse* of a in $\mathbb{Z}_n$.

We usually write "a inverse" as a^{-1}.
Inverses are **not** guaranteed to exist.

Theorem 10:
a has an inverse in $\mathbb{Z}_n$ if and only if $\gcd(a, n) = 1$

Problem 11:
Find the inverse of 3 in $\mathbb{Z}_4$, if one exists.
Find the inverse of 20 in $\mathbb{Z}_{14}$, if one exists.
Find the inverse of 4 in $\mathbb{Z}_7$, if one exists.

Solution

- 3^{-1} in $\mathbb{Z}_4$ is 3
- 20^{-1} in $\mathbb{Z}_{14}$ doesn't exist.
- 4^{-1} in $\mathbb{Z}_7$ is 2

Problem 12:
Show that if n is prime, every element of $\mathbb{Z}_n$ (except 0) has an inverse.

Problem 13:
Show that if n is not prime, $\mathbb{Z}_n$ has at least one element with no inverse.

Problem 14:
In general, how can we find the inverse of a in $\mathbb{Z}_n$? Assume a and n are coprime.
Hint: You can find that 34^{-1} is -175 in $\mathbb{Z}_{541}$ by looking at a previous problem.

Solution

We need an a^{-1} so that $a \times a^{-1} = 1$.
This means that $aa^{-1} - mk = 1$.
Since a and m are coprime, $\gcd(a, m) = 1$ and $aa^{-1} - mk = \gcd(a, m)$
Now use the extended Euclidean algorithm from Problem 6 to find $a^\star$.

Definition 15:
Elements in $\mathbb{Z}_n$ that have an inverse are called *units*.
The set of units in $\mathbb{Z}_n$ is called $\mathbb{Z}_n^\times$, which is read "$\mathbb{Z}$ mod n cross".

Problem 16:
What is $\mathbb{Z}_5^\times$?
What is $\mathbb{Z}_{12}^\times$?

Part 3: Groups

Group theory gives us a set tools for understanding complex structures. We can use groups to solve the Rubik's cube, to solve problems in physics and chemistry, and to understand complex geometric symmetries. It's also worth noting that much of modern cryptography is built using results from group theory.

Definition 17:
A *group* $(G, *)$ consists of a set G and an operator $*$.
Groups always have the following properties:

- **A:** G is closed under $*$. In other words, $a, b \in G \implies a * b \in G$.
- **B:** $*$ is associative: $(a * b) * c = a * (b * c)$ for all $a, b, c \in G$
- **C:** There is an *identity* $e \in G$, so that $a * e = a * e = a$ for all $a \in G$.
- **D:** For any $a \in G$, there exists a $b \in G$ so that $a * b = b * a = e$. b is called the *inverse* of a. This element is written as $-a$ if our operator is addition and a^{-1} otherwise.

Any pair $(G, *)$ that satisfies these properties is a group.

Problem 18:
Is $(\mathbb{Z}_5, +)$ a group?
Is $(\mathbb{Z}_5, -)$ a group?
Hint: $+$ and $-$ refer to the usual operations in modular arithmetic.

Problem 19:
Show that $(\mathbb{R}, \times)$ is not a group, then find a subset S of $\mathbb{R}$ so that $(S, \times)$ is a group.

Solution

$(\mathbb{R}, \times)$ is not a group because 0 has no inverse.
The solution is simple: remove the problem.

$(\mathbb{R} - \{0\}, \times)$ is a group.

Problem 20:
What is the smallest group we can create?

Solution

Let $(G, \circledcirc)$ be our group, where $G = \{\star\}$ and $\circledcirc$ is defined by the identity $\star \circledcirc \star = \star$
Verifying that the trivial group is a group is trivial.

Problem 21:
Let $(G, *)$ be a group with finitely many elements, and let $a \in G$.
Show that there exists an n in $\mathbb{Z}^+$ so that $a^n = e$
Hint: $a^n \coloneqq a * a * ... * a$, with a repeated n times.

The smallest such n defines the *order* of g.

Problem 22:
What is the order of 5 in $(\mathbb{Z}_{25}, +)$?
What is the order of 2 in $(\mathbb{Z}_{17}^{\times}, \times)$?

Theorem 23:
Let p be a prime number.
In any group $(\mathbb{Z}_p^{\times}, *)$ there exists a $g \in \mathbb{Z}_p^{\times}$ where...

- The order of g is $p - 1$, and
- $\{a^0,\ a^1,\ ...,\ a^{p-2}\} = \mathbb{Z}_n^{\times}$

We call such a g a *generator*, since its powers generate every other element in the group.

Note for Instructors

$\mathbb{Z}_p^{\times}$ has $p - 1$ elements.
The set $\{a^0,\ a^1,\ ...,\ a^{p-2}\}$ also has $p - 1$ elements, since we start counting from zero.

The fact that the last power here is $p - 2$ can be a bit confusing, but it's just the result of counting from zero. We could also write this set as $\{a^1,\ a^2,\ ...,\ a^{p-1}\}$, since $a^0 = a^{p-1}$.

Part 4: The Discrete Log Problem

Definition 24:
Let g be a generator in $(\mathbb{Z}_p^\times, *)$
Let n be a positive integer.

We now want a function "log" from $\mathbb{Z}_p^\times$ to $\mathbb{Z}^+$ so that $\log_g(g^n) = n$.
In other words, we want an inverse of the "exponent" function.

This is the *discrete logarithm problem*, often abbreviated *DLP*.

Problem 25:
Does the discrete log function even exist?
Show that exp is a bijection, which will guarantee the existence of log.
Note: Why does this guarantee the existence of log? Recall our lesson on functions.

Problem 26:
Find a simple (but perhaps inefficient) way to calculate $\log_g(a)$

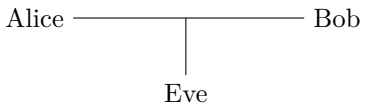
Problem 27:
Find an efficient way to solve the discrete log problem.
Then learn LaTeX, write a paper, and enjoy free admission to the graduate program at any university.

The discrete logarithm can be quickly computed in a few special cases, but there is no known way to efficiently compute it in general. Interestingly enough, we haven't been able to prove that an efficient solution *doesn't* exist. The best we can offer is a "proof by effort:" many smart people have been trying for long time and haven't solved it yet. It probably doesn't exist.

In the next few pages, we'll see how the assumption "DLP is hard" can be used to construct various tools used to secure communications.

Part 5: Diffie-Hellman Key Exchange

One problem we encounter in computer science is *secure key exchange*: How can two parties (usually called Alice and Bob) agree on a "key" without revealing anything to an eavesdropper (Eve)?



A simple mathematical solution to the key exchange problem is the *Diffie-Hellman key exchange algorithm*, detailed below.

Values that are *public* are known to everyone. Values that are sent are also known to everyone: we assume that everyone can see what Alice and Bob send to each other.
Eve can read all public values, but she cannot change them in any way.

Setup

Let p be a prime number
Let g be a generator in $\mathbb{Z}_p^\times$
Both g and p are public.

Alice	**Public**	**Bob**
Pick a random $a \in \mathbb{Z}_p^\times$	p, g	Pick a random $b \in \mathbb{Z}_p^\times$
Set $A = g^a$		Set $B = g^b$
Publish A	$A \longrightarrow$	
	$\longleftarrow B$	Publish B
Compute ...		Compute ...

Problem 28:
Complete the algorithm. What should Alice and Bob compute?
Hint: The goal of this process is to arrive at a *shared secret*
That is, Alice and Bob should arrive at the same value without exposing it to Eve.

Problem 29:
Let $p = 11$, $g = 2$, $a = 9$, and $b = 4$.
Run the algorithm. What is the resulting shared secret?

Solution

$g^b = 5$
$g^a = 6$
$g^{ab} = g^{ba} = 9$

Problem 30:
Is the Diffie-Hellman key exchange algorithm secure? What information does Eve have? What does Eve need to do to find the value Alice and Bob agreed on?

Problem 31:
Now, say Eve can change information in transit.
That is, she can pretend to be Alice to send information to Bob.
How can she break this system?
Note: This is called a *man-in-the-middle* attack.

Part 6: Elgamal Asymmetric Encryption

Another cryptographic tool we often use is the *public key cryptosystem*. In such a system, one has two keys: a *public key* that can only encrypt data, and a *private key* that can decrypt it. The following problem provides a simple example.

Problem 32:
Alice wants to send a secret letter to Bob. Eve, the postman, would like to see what is inside.

Alice has a box, a lock, and a key. Bob does not own a lock.
Eve will open the box if she can, but she will not try to break any locks.
Also, she will always deliver the box without modifying its contents.

How can Alice send her letter without letting Eve read it?

Elgamal encryption allows Alice to publish a public key (A in the diagram below), which Bob can use to encrypt a message. Alice then uses here private key (a) to decrypt it.

Setup

Let p be a prime number
Let g be a generator in $\mathbb{Z}_p^\times$
Both g and p are public.

Alice	Public	Bob
Pick a random $a \in \mathbb{Z}_p^\times$	p, g	Bob has a message $m \in \mathbb{Z}_p^\times$
Set $A = g^a$		Pick a random $k \in \mathbb{Z}_p^\times$
Publish A $\longrightarrow$	A $\longrightarrow$	
		Set $c_1 = g^k$
		Set $c_2 = mA^k$
$\longleftarrow$	(c_1, c_2) $\longleftarrow$	Publish (c_1, c_2)
Compute $c_2 \times c_1^{-a}$		
$= (mA^k)(g^{-ak})$		
$= (m)(g^{ak}g^{-ak})$		
$= m$		

Problem 33:
Let $p = 17$, $g = 2$, $a = 7$, $k = 10$, and $m = 3$
Run this algorithm and make sure it works.

Solution

$A = 2^7 = 9$
$c_1 = 2^10 = 4$
$c_2 = 3(9^{10}) = 5$

$c_1^a = 13$, so $c_1^{-a} = 4$
$c_2 \times c_1^a = 5 \times 4 = 3 = m$

Problem 34:
What information does Eve have?
What does Eve need to do to find m?

Problem 35:
Say Bob re-uses the same k twice.
Let (c_1, c_2) and (d_1, d_2) be two ciphertexts generated with this key, encrypting messages m_1 and m_2.
Also, say Eve knows the value of $m_1 - m_2$. How can Eve find m_1 and m_2?
Note: If Bob doesn't change his key, Eve will also be able to decrypt future messages.

Solution

$c_2 - d_2 = (m_1 - m_2)A^k$
So, $(c_2 - d_2)(m_1 - m_2)^{-1} = A^k$
Now that we have A^k, we can compute $m_1 = c_2 \times A^{-k}$

Part 7: Bonus Problems

Problem 36:
Show that a group has exactly one identity element.

Problem 37:
Show that each element in a group has exactly one inverse.

Problem 38:
Let $(G, *)$ be a group and $a, b, c \in G$. Show that...

- $a * b = a * c \implies b = c$
- $b * a = c * a \implies b = c$

This means that we can "cancel" operations in groups, much like we do in algebra.

Problem 39:
Let G be the set of all bijections $A \to A$.
Let $\circ$ be the usual composition operator.
Is $(G, \circ)$ a group?

Definition 40:
Note that our definition of a group does **not** state that $a * b = b * a$.
Many interesting groups do not have this property. Those that do are called *abelian* groups.

One example of a non-abelian group is the set of invertible 2x2 matrices under matrix multiplication.

Problem 41:
Show that if G has four elements, $(G, *)$ is abelian.

Problem 42:
Prove Theorem 10:
a has an inverse mod m iff $\gcd(a, m) = 1$

Solution

Assume $a^\star$ is the inverse of $a \pmod m$.
Then $a^\star \times a \equiv 1 \pmod m$
Therefore, $aa^\star - 1 = km$, and $aa^\star - km = 1$
We know that $\gcd(a, m)$ divides a and m, therefore $\gcd(a, m)$ must divide 1.
$\gcd(a, m) = 1$
Now, assume $\gcd(a, m) = 1$.
By the Extended Euclidean Algorithm, we can find (u, v) that satisfy $au + mv = 1$
So, $au - 1 = mv$.
m divides $au - 1$, so $au \equiv 1 \pmod m$
u is $a^\star$.

Problem 43:
The Euclidean Algorithm (From Problem 5) can be written as follows:

- Assume $a > b$.
- Set $e_0 = a$ and $e_1 = b$.
- Let $e_{n+1} = \text{remainder}(r_{n-1} \div r_n)$
- Stop when $e_k = 0$.
- Then, $\gcd(a, b) = e_{k-1}$.

Let F_n be the n^{th} Fibonacci number. ($F_0 = 0$; $F_1 = 1$; $F_2 = 1$; ...)

Show that if the Euclidean algorithm requires n steps for an input (a, b), then $a \geq F_{n+2}$ and $b \geq F_{n+1}$. In other words, show that the longest-running input of a given size is a Fibonacci pair.

Solution

The easiest way to go about this is induction on n:
Base Case:
If $n = 1$, b divides a with no remainder, and the smallest possible a, b for which this is true is $(2, 1) = (F_3, F_2)$.

Induction:
Assume that for n steps, $a \geq F_{n+2}$ and $b \geq F_{n+1}$.
Now, say the algorithm takes $n + 1 = m$ steps.
The first step gives us $a = q_0 b + r_0$
Therefore, the pair (b, r_0) must take $m - 1$ steps.
We thus know that $b \geq F_{m+1}$ and $r_0 \geq F_m$ by our induction hypothesis
Therefore, $a = q_0 b + r_0 \geq b + r_0$
But $b + r_0 = F_{m+1} + F_m = F_{m+2}$,
so $a \geq F_{m+2}$.

Problem 44: Chinese Remainder Theorem
There are certain things whose number is unknown. If we count them by threes, we have two left over; by fives, we have three left over; and by sevens, two are left over. How many things are there?

Solution

$x \equiv 2 \pmod 3$
$x \equiv 3 \pmod 5$
$x \equiv 2 \pmod 7$
$x = 23 + 105k \ \forall k \in \mathbb{Z}$

Problem 45:
Show that if p is prime, $\binom{p}{i} \equiv 0 \pmod p$ for $0 < i < p$.

Solution

$\binom{p}{i} = \frac{p!}{i!(p-i)!}$ tells us that $i!(p-i)!$ divides $p! = p(p-1)!$.
However, $i!(p-i)!$ and p are coprime, since all factors of $i!(p-i)!$ are smaller than p.
Therefore, $i!(p-i)!$ must divide $(p-1)!$
So, $\binom{p}{i} = p \times \frac{(p-1)!}{i!(p-i)!}$, and $\binom{p}{i} \equiv 0 \pmod p$.

Problem 46: Fermat's Little Theorem
Show that if p is prime and $a \not\equiv 0 \pmod{p}$, then $a^{p-1} \equiv 1 \pmod{p}$.
You may want to use Problem 45.
Hint: It may be easier to show that $a^p \equiv a \pmod{p}$

Solution

Use induction:
$1 \equiv 1 \pmod{p}$
Using Problem 45 and the binomial theorem, we have
$2^p = (1+1)^p = 1 + \binom{p}{1} + \binom{p}{2} + \cdots + \binom{p}{p-1} + 1 \equiv 1 + 0 + ... + 0 + 1 \equiv 2 \pmod{p}$
Then,
$3^p = (1+2)^p = 1 + \binom{p}{1}2 + \binom{p}{2}2^2 + \cdots + \binom{p}{p-1}2^{p-1} + 2^p \equiv 1 + 0 + ... + 0 + 2 \equiv 3 \pmod{p}$
We can repeat this for all a. This proof can be presented more formally with a bit of induction.

Problem 47:
Show that for any three integers a, b, c,
$\gcd(ac + b, a) = \gcd(a, b)$

[Note on Problem 43] This proof can be used to show that the Euclidean algorithm finishes in logarithmic time, and it is the first practical application of the Fibonacci numbers. If you have finished all challenge problems, finish the proof: find how many steps the Euclidean algorithm needs to arrive at a solution for a given a and b.