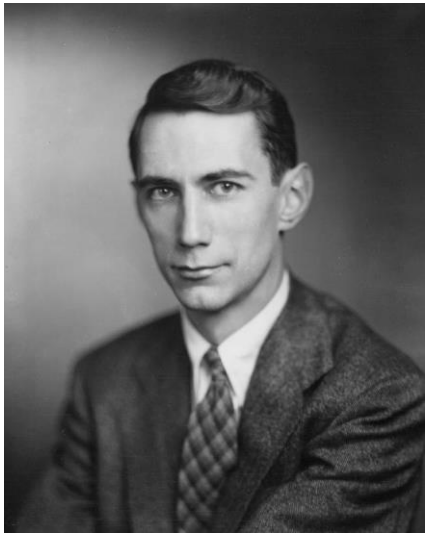
COMM066

Mathematical Foundations of Cryptography - some pointers and exercises

Dr. Robert Granger
Surrey Centre for Cyber Security
School of Computer Science and Electronic Engineering

Motivation – why so much mathematics?



`A Mathematical Theory of Cryptography‘

Claude E. Shannon, 1945.

Modern cryptography relies on mathematics, computer science, information theory, computational complexity theory, software engineering, proof techniques, formal verification and more...

Motivation – why so much mathematics?

In this module you will need **solid foundations** upon which to build your understanding. In particular, you should be familiar with the following mathematical concepts:

- Bits, bitstrings and Boolean functions
- Logic and proofs, proof techniques
- Sets and numbers
- Algorithms and their complexity
- Discrete probability theory

The main reference is the book `Discrete Mathematics and Its Applications' by Kenneth H. Rosen (see Course Materials > Mathematics Background Material > this book). The most relevant chapters for our module are Chapters 1 to 7.

Rosen – please review your familiarity

Bits and Boolean functions (Section 1.1.6) - bit, Binary field $GF(2) = \{0,1\}$, XOR, AND, NOT, bit strings, bitwise operations, Boolean functions.

Basic proof techniques (Section 1.7/1.8): Direct proof of a conditional statement $p \rightarrow q$, Proof by contraposition, the statement $p \rightarrow q$ is equivalent to $\neg q \rightarrow \neg p$, Proof by contradiction. Exhaustive proof and Proof by cases, Looking for Counterexamples, Proof by induction.

Sets and Numbers (Chapters 2 and 4): Modular arithmetic: a (mod n), Numbers in base 2 and base 16, Binomial coefficient $\binom{n}{k}$

Algorithms (Chapter 3): Growth of functions, Complexity of algorithms

Discrete Probability (Chapter 7): Random variable, Probability distribution, Conditional probability, Expectation & Variance, Uniform & Bernoulli distribution

Motivation – why so much mathematics?

Famous public-key cryptosystems

Diffie-Hellman Key Agreement (1976)	$(\mathbb{Z}/p\mathbb{Z})^*$
RSA Encryption/Signatures (1978)	$(\mathbb{Z}/N\mathbb{Z})^*$ where $N = pq$
Elliptic Curve Cryptography (1985)	$\{(x,y) \in (\mathbb{Z}/p\mathbb{Z}) \times (\mathbb{Z}/p\mathbb{Z}): y^2 = x^3 + ax + b\} \cup \{\infty\}$

Post-quantum cryptography can be based on:

- Lattice-based Cryptography
- Multivariate system-based Cryptography
- Code-based Cryptography
- Isogeny-based Cryptography

We also have:

- Secure Multiparty Computation
- Fully Homomorphic Encryption
- Secret Sharing Schemes... And many more...

All are based on (fairly natural) ***computational mathematical problems.***

Motivation – why this foundations tutorial?

In **Lecture 6** on Number Theory and Hard Problems

(the start of public-key or Asymmetric Cryptography) you will see:

- Divisibility and Euclid's Algorithm
- Modular Arithmetic
- Integer Factorisation Problem
- Group theory, cyclic groups and generators
- Discrete Logarithm Problem

Facility with the techniques is so important to the subject and the assessment strategy: we typically spend 2 to 3 labs strengthening skills.

Today's Agenda (thanks to Laurian, Postdoctoral Fellow in Cryptography):
- Some instructive discrete maths exercises to try, together with solutions
- A short lecture on the first topic above, with examples and exercises.

Divisibility and Euclid's Algorithm

Number Sets and Binary Operations

Number Sets

$\mathbb{N}$	$= \{1, 2, 3, \ldots.\}$	positive integers (natural numbers)
$\mathbb{N}_0$	$= \{0, 1, 2, \ldots.\}$	non-negative integers ($\mathbb{N}$ & 0)
$\mathbb{Z}$	$= \{0, \pm1, \pm2, \ldots\}$	integers
$\mathbb{Q}$	$= \{a/b \mid \forall\, a \in \mathbb{Z}, b \in \mathbb{Z} \backslash 0\}$	rational numbers
$\mathbb{R}$		real numbers

Binary Operation $\circ : S \times S \longrightarrow S$ given $a, b \in S$ it returns $a \circ b$

associative	$\forall\, a, b, c \in S$	$(a \circ b) \circ c = a \circ (b \circ c)$
commutative	$\forall\, a, b \in S$	$a \circ b = b \circ a$

Let $S' \subset S$.

S' is closed under $\circ$	$\forall\, a, b \in S'$	$a \circ b \in S'$

Primes and Composites

Primes and Composites

Let $N \in \mathbb{N}$.

N is called prime if its only positive divisors are 1 and N;
else N is called composite.

Theorem (Factorisation - Euclid)

Every positive integer $N \in \mathbb{N}$ can be uniquely expressed as

$$N = P_1^{e1} \cdots P_r^{er}$$

where

$P_1, \ldots, P_r$ are distinct primes

$e_1, \ldots, e_r$ are positive integers.

Example

$N = 90$

$= 2 \cdot 3^2 \cdot 5$

Factorisation Examples

$$90 = 2 \cdot 3^2 \cdot 5$$

$$1024 = 2^{10}$$

$$41067 = 3^5 \cdot 13^2$$

$$56700 = 2^2 \cdot 3^4 \cdot 5^2 \cdot 7$$

$$1234567 = 127 \cdot 9721$$

$$3335750811866041 = 42356411 \cdot 78754331$$

Divisibility

Divisibility Let $a, b \in \mathbb{Z}$.

- We say a is a divisor of b iff $b = az$ for some $z \in \mathbb{Z}$
- $a|b \stackrel{\text{def}}{=}$ a is a divisor of b, or a divides b
- if a does not divide b then we write $a \nmid b$

Theorem For all $a, b, c \in \mathbb{Z}$ we have

- $a|a$, $1|a$, and $a|0$
- $0|a$ if and only if $a = 0$
- $a|b$ and $a|c$ implies $a|(b+c)$
- $a|b$ and $b|c$ implies $a|c$

Greatest Common Divisor

Theorem (Divisibility with Remainder)

Let $a, b \in \mathbb{Z}$ with $b > 0$.

There exist unique $q, r \in \mathbb{Z}$ such that $a = bq + r$ and $0 \leq r < b$.

Example

$26 = 7 \cdot 3 + 5$

$a = 26, b = 7$

$q = 3, \quad r = 5$

q is called the quotient, r is called the remainder.

Greatest Common Divisor $d = \gcd(a,b)$ (sometimes just (a,b))

- d is the largest integer such that $d|a$ and $d|b$.
- If $d = 1$ then a and b are called relatively prime or co-prime.

Example

$\gcd(18, 12) = 6$

$\gcd(13, 27) = 1$

GCD Theorem

Let $a, b \in \mathbb{Z}$.

There exist $x, y \in \mathbb{Z}$ such that $xa + yb = d$ where $d = \gcd(a, b)$ and

d is the smallest positive integer that can be expressed in this way.

Implications of the GCD Theorem

Let $a, b, c \in \mathbb{Z}$.

Fact (Exercise in Lab 6)
If $c|ab$ and $\gcd(a,c) = 1$ then $c|b$.

If p is prime and $p|ab$ then either $p|a$ or $p|b$.

Example $5|7 \cdot 10$, $\gcd(5, 7) = 1$, $5|10$

Fact (Exercise in Lab 6)
If $a|c$ and $b|c$ and $\gcd(a, b) = 1$ then $ab|c$.

Example $5|70$ and $7|70$, $\gcd(5,7) = 1$, $35|70$

How should one compute gcd(a,b)?

Without loss of generality we may assume $a > b > 0$.

Option 1 (naive):
Find the prime factorisations of a and of b. Then for each prime that divides both a and b take the minimum exponent and form the product over all such primes.

Option 2 (Euclid's ingenious observation, 300BC):
Since $d|a$ and $d|b$ we have $d|(a - qb)$ for any $q \in \mathbb{Z}$.

If we choose q to be the quotient and r the remainder then $d|r$ and $0 \leq r < b$.

Hence $\gcd(a,b) = \gcd(b,r)$ and $a > b > r \geq 0$.

REPEAT UNTIL $r = 0$ and then gcd is the previous remainder.

Example gcd computations using Euclid

Example 1: compute gcd(100,67)

$100 = 67 \cdot 1 + 33 \quad (q = 1, r = 33)$
$67 = 33 \cdot 2 + 1 \quad (q = 2, r = 1)$
$33 = 1 \cdot 33 + 0 \quad (q = 33, r = 0)$
So $\gcd(100,67) = \gcd(33,1) = 1$.

Example 2: compute gcd(101,38)

$101 = 38 \cdot 2 + 25 \quad (q = 2, r = 25)$
$38 = 25 \cdot 1 + 13 \quad (q = 1, r = 13)$
$25 = 13 \cdot 1 + 12 \quad (q = 1, r = 12)$
$13 = 12 \cdot 1 + 1 \quad (q = 1, r = 1)$
$12 = 1 \cdot 12 + 0 \quad (q = 12, r = 0)$
So $\gcd(101,38) = \gcd(12,1) = 1$.

Example gcd computations using Euclid

Example 3: compute gcd(102,17)

$102 = 17 \cdot 6 + 0 \quad (q = 6, r = 0)$

So gcd(102,17) = 17 (we treat 17 as the previous remainder).

Example 4: compute gcd(102,18)

$$\begin{aligned} 102 &= 18 \cdot 5 + 12 \quad (q = 5, r = 12) \\ 18 &= 12 \cdot 1 + 6 \quad (q = 1, r = 6) \\ 12 &= 6 \cdot 2 + 0 \quad (q = 2, r = 0) \end{aligned}$$

So gcd(102,18) = gcd(12,6) = 6.

Extended Euclidean Algorithm

Recall if $a, b \in \mathbb{Z}$ then $\exists\, x, y \in \mathbb{Z}$ with $xa + yb = \gcd(a, b) = d$

Extended Euclidian Algorithm eGCD(a, b) returns (d,x,y)

Idea: rewind the gcd computation (Note: a recursive algorithm is given in Lecture 6)

Example 1: compute eGCD(100,67). Recall

$100 = 67 \cdot 1 + 33$
$67 = 33 \cdot 2 + 1$
$33 = 1 \cdot 33 + 0$

So gcd(100,67) = 1. Starting with the second to last line, we express the gcd as

$1 = 67 - 33 \cdot 2$
$= 67 - 2 \cdot (100 - 67 \cdot 1)$ (using 1st equation, eliminating 33)
$= -2 \cdot 100 + 3 \cdot 67$

So (d,x,y) = (1,-2,3).

Extended Euclidean Algorithm

Example 2: compute eGCD(101,38). Recall

$101 = 38 \cdot 2 + 25$
$38 = 25 \cdot 1 + 13$
$25 = 13 \cdot 1 + 12$
$13 = 12 \cdot 1 + 1$
$12 = 1 \cdot 12 + 0$

So gcd(101,38) = 1. Starting with the second to last line, we express the gcd as

$1 = 13 - 12 \cdot 1$
$= 13 - 1 \cdot (25 - 1 \cdot 13)$ (using 3rd equation, eliminating 12)
$= -1 \cdot 25 + 2 \cdot 13$
$= -1 \cdot 25 + 2 \cdot (38 - 1 \cdot 25)$ (using the 2nd equation, eliminating 13)
$= 2 \cdot 38 - 3 \cdot 25$
$= 2 \cdot 38 - 3 \cdot (101 - 2 \cdot 38)$ (using the 1st equation, eliminating 25)
$= -3 \cdot 101 + 8 \cdot 38$

So (d,x,y) = (1,-3,8).