4. Number Theory and Cryptography

4.1: Divisibility and Modular Arithmetic

Divisibility and Its Properties

Divisibility is fundamental in number theory. If a and b are integers with $a \neq 0$, then a divides b (denoted $a|b$) if there exists an integer c such that:

$b=ac$

For example, $5|20$ because $20=5\times4$, but $5\nmid 22$ since 22 is not a multiple of 5.

Properties of Divisibility:

- If $a|b$ and $a|c$, then $a|(b+c)$.
- If $a|b$, then $a|bc$ for any integer c.
- If $a|b$ and $b|c$, then $a|c$.

The Division Algorithm

For any integer a and positive integer d, there exist unique integers q (quotient) and r (remainder) such that:

$a=dq+r,\ 0\leq r<d$

Example:
Dividing 101 by 11:

$101=11\times9+2$

Here, $q=9$ and $r=2$.

Modular Arithmetic

In modular arithmetic, two integers are **congruent modulo m** if they have the same remainder when divided by m:

$a\equiv b \pmod m$ if and only if $m|(a-b)$

Example:

$29\equiv5 \pmod{12}$

because $29-5=24$, which is divisible by 12.

Properties of Modular Arithmetic:

1. $(a+c)\equiv(b+d) \pmod m$
2. $(a\cdot c)\equiv(b\cdot d) \pmod m$

4.2: Integer Representations and Algorithms

Base b Expansion

Any integer n can be expressed in base b as:

$n = a_k b^k + a_{k-1} b^{k-1} + ... + a_1 b + a_0$

where $0 \le a_i < b$.

Example:
Converting 241 to binary:

$(241)_{10} = (11110001)_2$

Algorithms for Base Conversion

To convert from decimal to base b, repeatedly divide by b, recording the remainders.

Binary Arithmetic

Binary addition follows:

- 0+0=0
- 0+1=1
- 1+1=10(carry 1)

Multiplication is performed using shift-and-add methods.

Modular Exponentiation

Used in cryptography for fast exponentiation:

$b^n \bmod m$

is computed efficiently by **exponentiation by squaring**.

4.3: Primes and Greatest Common Divisors

Prime Numbers and Factorization

A prime number is a number greater than 1 with exactly two distinct divisors: 1 and itself.

Prime Factorization Example:

$120 = 2^3 \times 3 \times 5$

Greatest Common Divisor (GCD)

The GCD of a and b is the largest number dividing both.

Using the **Euclidean Algorithm**:

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

Example:

gcd(252,198)=18

Bézout's Identity

For any integers a and b, there exist integers x and y such that:

$\gcd(a,b)=ax+by$

4.4: Solving Congruences

Linear Congruences

Equations of the form:

$ax \equiv b \pmod{m}$ have a solution if $\gcd(a,m) \mid b$.

The Chinese Remainder Theorem (CRT)

If we have:

$x \equiv a_1 \pmod{m_1}, x \equiv a_2 \pmod{m_2}$

where m_1 and m_2 are coprime, then there is a unique solution modulo $m_1 m_2$.

4.5: Applications of Congruences

Hashing Functions

Used to store data efficiently:

$h(k)=k \bmod m$

Pseudorandom Number Generation

Using the **linear congruential generator (LCG)**:

$x_{n+1}=(ax_n+c) \bmod m$

Check Digit Schemes

Used in ISBN and credit card validation.

Example (ISSN Checksum):

$d_8 \equiv 3d_1+4d_2+5d_3+\cdots+9d_7 \pmod{11}$

4.6: Cryptography

Classical Ciphers

The **Caesar Cipher** shifts letters by kkk:

$f(p)=(p+k) \bmod 26$

Public-Key Cryptography (RSA)

1. Choose primes p and q, compute $n=pq$.
2. Compute $\varphi(n)=(p-1)(q-1)$.
3. Choose e with $\gcd(e,\varphi(n))=1$.
4. Compute $d \equiv e^{-1} (\bmod \varphi(n))$.
5. Encrypt: $c=m^e \bmod n$.
6. Decrypt: $m=c^d \bmod n$.

Diffie-Hellman Key Exchange

Allows two parties to share a secret key securely.

Homomorphic Encryption

Allows computations on encrypted data, useful for **secure cloud computing**.