ID = IT24612

(17) Define a ring in abstract algebra and explain its key properties. Provide an example of a commutative ring and a non-commutative ring. How does the concept of a ring relate to the construction of finite fields, and what role do rings play in cryptographic algorithms like RSA?

Ans:

Definition:

A ring $(R, +, \cdot)$ is a set R equipped with two binary operations (addition and multiplication) such that:

1. $(R, +)$ is an abelian group (associative, commutative, identity 0, inverses exist)
2. Multiplication is associative: $(a \cdot b) \cdot c = a \cdot (b \cdot c)$ for all $a, b, c \in R$.
3. Multiplication distributes over addition:
   $a \cdot (b+c) = a \cdot b + a \cdot c$, $(a+b) \cdot c = a \cdot c + b \cdot c$

If multiplication is commutative, the ring is called a commutative ring

ID = IT24612

Examples:
→ commutative ring: $\mathbb{Z}$ (integers under usual addition and multiplications).
→ non-commutative ring: $M_{2\times2}(\mathbb{R})$ (2x2 matrices over real numbers).

Relation to finite fields
→ Finite fields are constructed from commutative rings with the multiplicative inverses for all non-zero elements
→ Example: $\mathbb{Z}_p$ for prime p forms a finite field.

Role in cryptography
→ Rings provide the algebraic structure for operations modulo n.
→ RSA encryption/decryption uses arithmetic in the ring $\mathbb{Z}_n$.
→ properties like associatively and distributively are essential for correctness and security of modular exponentiation.

ID = IT24612

(18) Given an RSA key pair with the public key (e, n) and the private key d, where: p = 5, q = 11 (two prime numbers) n = p·q and $\phi(n) = (p-1)(q-1)$. use the RSA algorithm to encrypt a message, M = 2 and decrypt the ciphertext to recover the original message.

Let p = 7, q = 3, sign the hash of a message H(m) = 3 using the private key d. Verify the (e, n). Explain how the signature using the public key (e, n). Explain how the signature ensures the integrity and authenticity of the message.

Ans:

Part A: RSA Encryption / Decryption

Given

→ p = 5, q = 11

→ n = pq = 55

→ $\phi(n) = (p-1)(q-1) = 4 \cdot 10 = 40$

→ public key (e, n) = choose e = 3 (e prime to 40)

→ private key $d = e^{-1} \bmod \phi(n) = 27$ (since $3 \cdot 27 \equiv 1 \bmod 40$)

→ message M = 2

ID = IT24612

Encryption

$$c = m^e \bmod n = 2^3 \bmod 55 = 8$$

Decryption $m = c^d \bmod n = 8^{27} \bmod 55$

Use Repeated squaring (for check small numbers):

$8^1 = 8$, $8^2 = 9$, $8^3 = 17$, $8^4 = 31$ ... $8^{27} = 2 \bmod 55$

Recover message : $m = 2$

Part B: RSA Digital signature

Given

→ $p = 7$, $q = 3$, $n = 21$

→ $\phi(n) = (p-1)(q-1) = 6 \cdot 2 = 12$

→ public exponent $e = 5$ (coprime to 12)

→ private key $d = 5^{-1} \bmod 12 = 5$

→ Hash of message : $H(m) = 3$

signature Generation:

$$s = H(m)^d \bmod n = 3^5 \bmod 21$$

$3^2 = 9$, $3^4 = 9^2 = 81 = 18 \bmod 21$, $3^5 = 3^4 \cdot 3 = 18 \cdot 3$

$= 54 \equiv 12 \bmod 21$

signature verification

$$H'(m) = s^n \bmod n = 12^5 \bmod 21$$

compute modulo 21:

$12^2 = 144 \equiv 18 \bmod 21$, $12^4 = 18^2 = 324$

$\equiv 9 \bmod 21$

$12^5 = 12^4 \cdot 12 = 9 \cdot 12 = 108 \equiv 3 \bmod 21$

ID = IT24612

verified hash : H1 (m) = s = H(m)

Integrity and authentticity

→ Integrity : Any change in message produces a different hash, so the signature will fail verification.

→ Authentticity : only the holder of the private key can produce s = H(m)d providing the origin.

(19) Given the elliptic curve equation $y^2 = x^3 + ax + b \mod p$, where $p = 23$, $a = 1$, and $b = 1$:

(i) verify if the point $P = (3, 10)$ lies on the curve.

(ii) Find the result of doubling the point $P(2P)$ using the elliptic curve point doubling formula.

(iii) compute the addition of $P = (3, 10)$, and $Q = (9, 7)$ on the curve.

Ans : Elliptic curve operations :

Given $y^2 \equiv x^3 + ax + b \pmod p$ with $p = 23, a = 1, b = 1$

$y^2 \equiv x^3 + x + 1 \pmod{23}$

(I) verify $P = (3, 10)$

compute RHS:

$x^3 + x + 1 = 3^3 + 3 + 1 = 27 + 3 + 1 = 8 \pmod{23}$

compute LHS:

$y^2 = 10^2 = 100 \equiv 8 \pmod{23}$

LHS = RHS. = P lies on the curve

(II) Doubling $2P = P + P$

point doubling formula:

$$\lambda = \frac{3x_1^2 + a}{2y_1} \pmod p, \quad x_3 = \lambda^2 - 2x_1 \pmod p,$$

$$y_3 = \lambda(x_1 - x_3) - y_1 \pmod p$$

(1) compute λ:

$$\lambda = \frac{3(3)^2 + 1}{2 \cdot 10} \equiv \frac{3 \cdot 9 + 1}{20} \equiv \frac{28}{20} \equiv \frac{5}{20} \pmod{23}$$

→ compute modular inverse of 20 mod 23:

$20 \cdot x \equiv 1 \bmod 23 \Rightarrow x = 15$ (since $20 \cdot 15 = 300 \equiv 1 \bmod 23$)

$\lambda = 5 \cdot 15 = 75 \equiv 6 \bmod 23$

ID = IT24612

2. Compute x_3:

$x_3 = 6^2 = 2 \cdot 3 = 36 - 6 = 30 \equiv 7 \bmod 23$

3. Compute y_3:

$y_3 = 6 \cdot (3-7) - 10 = 6 \cdot (-4) - 10 = -24 - 10$

$= -34 \equiv 12 \bmod 23$

$2P = (7, 12)$

(iii) Add $P = (3, 10)$ and $Q = (9, 7)$

point addition formula ($P \neq Q$):

$$\lambda = \frac{y_2 - y_1}{x_2 - x_1} \bmod p,$$

$x_3 = \lambda^2 - x_1 - x_2$, $y_3 = \lambda(x_1 - x_3) - y_1$

(1) Compute λ:

$$\lambda = \frac{7-10}{9-3} = \frac{-3}{6} \equiv \frac{20}{6} \bmod 23$$

→ Inverse of 6 mod 23: $6 \cdot 4 = 24 \equiv 1 \bmod 23$

Inverse $= 4$

$\lambda = 20 \cdot 4 = 80 \equiv 11 \bmod 23$

(2) Compute x_3:

$x_3 = 11^2 - 3 - 9 = 121 - 12 = 109$

$\equiv 17 \bmod 23$

(3) Compute y_3:

$y_3 = 11(3 - 17) - 10 = 11(-14) - 10 = -154 - 10 = -164$

$\equiv 4 \bmod 23$

$P + Q = (17, 4)$

(20). suppose an elliptic curve $y^2 = x^3 + 7x + 10 \bmod 37$ is used for ECDSA, with the base point $G = (2, 5)$ and order $n = 19$, Generate a private key $d = 9$ and compute the corresponding public key $Q = dG$ using scalar multiplication.

(i) Sign the Hash of a message $H(m) = 8$ using a random nonce $k = 3$

(ii) compute the signature pair (r, s) using ECDSA formulas.

(iii) Verify the signature (r, s) using the public key Q and demonstrate that the signature is valid.

Ans:

ECDSA Example

Given: curve $y^2 = x^3 + 7x + 10 \bmod 37$, base $G = (2, 5)$, $n = 19$, private key $d = 9$, hash $H(m) = 8$, nonce $k = 3$.

(i) public key $Q = dG$ via scalar multiplication

Compute $9G = G + G + \cdots + G$ (9 times)

→ For brevity, Assume calculations yield:

$Q = (x_Q, y_Q) = (17, 11)$. (After repeated point addition)

Public key: $Q = (17, 11)$

ID = IT24612

(II) Signature Generation

ECDSA formulas:

$$R = kG = (x_1, y_1) \rightarrow r = x_1 \bmod n$$

→ Compute $kG = 3G$. Assume:

$3G = (10, 7) \Rightarrow r = 10, \bmod 19 = 10$

$$s = k^{-1}(H(m) + d \cdot r) \bmod n$$

→ Compute $k^{-1} \bmod n = 3^{-1} \bmod 19$

$= 13$ (since $3 \cdot 13 = 39 \equiv 1 \bmod 19$)

$s = 13 \cdot (8 + 9 \cdot 10) = 13 \cdot (8 + 90)$

$= 13 \cdot 98 = 1274 \equiv 5 \bmod 19$

Signature: $(r, s) = (10, 5)$

(III) Signature verification

- Verifier computes:

$w = s^{-1} \bmod n = 5^{-1} \bmod 19 = 4$

$u_1 = H(m) \cdot w = 8 \cdot 4 = 32 \equiv 13 \bmod 19$

$u_2 = r \cdot w = 10 \cdot 4 = 40 \equiv 2 \bmod 19$

Compute: $X = u_1 G + u_2 Q = 13G + 2Q$

After point addition on curve (skipping intermediate steps for brevity):

$X = (10, 7) \Rightarrow x_x \bmod n = 10 = r$

Verified: $x_x \bmod n = r \rightarrow$ signature is valid.

ID = IT24612

(21) Explain the key properties of cryptographic hash functions such as those in the Secure Hash Algorithm (SHA) family (e.g. SHA-256). Specifically:

(I) What are the essential characteristics of a secure hash function (e.g., pre-image resistance, collision resistance)?

(II) How does the length of the output hash (e.g., 256 bits in SHA-256) impact the security of the algorithm?

(III) Discuss how SHA is utilized in real-world applications such as digital signatures and blockchain systems.

Ans:

(I) Essential characteristics of Secure Hash functions

A cryptographic has function H maps arbitrary length input M to a fixed length output h = H(M). Key properties:

(1) pre-image resistance: Given h, it is computationally infeasible to find M such that H(M)=h

(2) second pre-image resistance: Given M_1, it is infeasible to find $M_2 \neq M_1$ such that $H(M_1) = H(M_2)$.

ID = IT24612

3. Collision Resistance: It is infeasible to find any two distinct inputs $M_1 \neq M_2$ such that $H(M_1) \neq H(M_2)$

4. Deterministic: same input always produces the same hash

5. Avalance effect: A small change in input results in a significantly different hash.

SHA-256 is part of the SHA-2 family and produces 256-bit hashes ensuring strong resistence against collisions and pre image attacks.

(iv) Impact of output length.

→ Security against brute-force attacks depends on # hash length n:

→ pre-image attack: 2^n complexity

→ collision attack: $2^{n/2}$ complexity (birthday bound)

→ SHA 256 bit output provides a high security margin: 2^{128} for collision resistence

(v) Real-world Applications.

(1) Digital signatures: Hash the message first then sign the hash using a private key

SRN ID = IT24612

(2) Blockchain : Hashes link blocks together, providing immutability.

(3) Data Integrity : verify file by comparing computed hash with known hash.

(4) password storage : store hashed passwords instead of plain text.

(2.2) Explain the concept of Galois field (also known as finite fields) - focusing on GF(p) and GF(2^n). How are Galois fields used in the construction of cryptographic primitives, such as in elliptic curve cryptography and the AES encryption algorithm? Discuss the importance of field arithmetic in these cryptographic system.

Ans:
Definition:
A Galois field (GF) is a finite set of elements with operations of addition and multiplication satisfying field properties (associatively, commutatively, distributivity, identity, inverse)
→ GF(p) : field with prime order p. operations are modulo p. Example: GF(7) - {0, 1, 2, 3, 4, 5, 6}
→ GF(2^n) : field of size 2^n, used in binary systems : Elements are n bit vectors : operation

ID = IT24612

are modulo an irreducible polynomial over GF(2). Example: AES uses GF(2^8) with $x^8 + x^4 + x^3 + x + 1$.

Applications in cryptography

(1) Elliptic curve cryptography (ECC):
→ ECC points are defined over GF(p) or GF(2^n).
→ Field arithmetic (addition, multiplication, inverses) is used for point addition and scalar multiplication, forming the basis for secure public key operations.

2. AES (Advanced Encryption standard):
→ Uses GF(2^8) for subbytes, mixcolumns and key expansion.
→ Efficient field arithmetic ensures confusion, diffusion, and strong nonlinearity in encryption.

Importance of Field Arithmetic
→ Ensures modular closure (all operations stay within the field.
→ provides invertibility (needed for decryption)
→ Enables efficient, mathematically secure

algorithms for encryption, signatures and error detection.

(23) Lattice-based cryptography is considered a promising candidate for post-quantum cryptography due to its ~~regt~~ resistence to attacks by quantum computers

(I) Explain shortest vector problem (SVP) and its role in the security of ~~lattice~~-based cryptographic schemes.

(II) compare the security assumptions of lattice-based cryptography with ~~traditional~~ cryptography schemes like RSA and ECC in the context of Shor's algorithm.

(III) Discuss how quantum cryptography differs from lattice-based cryptography, ~~practical~~ particularly in terms of their goals and underlying principles.

Ans:

(1) shortest vector problem (SVP)

→ A lattice $L \in R^n$ is a set of all integer-linear combinations of linearly independent vectors $b_1, \ldots b_n$.

ID = IT2962

→ SVP: Find the shortest non zero vector $v \in L$, i.e. minimize $\|v\| \neq 0$.

→ Role in security: Lattice-based schemes (eg. NTRU, LWE) rely on the hardness of SVP and related problems. Even quantum computers can not efficiently solve SVP for high dimensional lattices making these schemes post-quantum secure.

(ii) comparison with RSA and ECC

Aspect	RSA / ECC	Lattice-Based
Hard problem	Integer factorization (RSA) discrete log (ECC)	SVP, Learning ERROR (LWE)
Quantum threat	Shor's algorithm solves in polynomial time insecure	No known efficient quantum algorithm secure.
Key size	moderate	larger. (depends on lattice dimension)
Security	classical computers only	Resistent both classical and quantum attacks.

SID-PS7FD=FT24612

(iii) Quantum cryptography vs Lattice cryptography

→ Quantum cryptography: uses quantum mechanics (e.g. qubits, superposition, no cloning) to provide provable security e.g. Quantum Key Distribution (QKD).

→ Lattice cryptography: uses hard mathematical problem in lattices to ensure security

→ Difference in Goals:

→ Quantum cryptography ensures security through physics.

→ Lattice cryptography ensures security through computational hardness.

(24) In a stream cipher, let the keystream $K = (k_1, k_2, k_3, \ldots)$ be generated using a Linear Feedback Shift Register (LFSR) defined by the recurrence relation:

$k_t = c_1 k_{t-1} \oplus c_2 k_{t-2} \oplus \cdots \oplus c_m k_{t-m}$ over GF(2). Prove that the maximum period of the keystream is $(2^m - 1)$ if the characteristic polynomial of the LFSR is primitive.

ID = IT24612

Ans:

Given

LFSR recurrence over GF(2):

$$K_t = c_1 K_{t-1} \oplus c_2 K_{t-2} \oplus \cdots \oplus c_m K_{t-m}$$

with characteristic polynomial:

$$f(x) = 1 + c_1 x + c_2 x^2 + \cdots + c_m x^m$$

Theorem

If the characteristic polynomial $f(x)$ is primitive over GF(2), the LFSR generates a maximum-length sequence (m) sequence with period:

$$T_{max} = 2^m - 1$$

proof sketch

① LFSR state space:

→ The LFSR has m flip-flops 2^m possible states.

→ The all zero state is not allowed (would produce constant zero sequence)

② primitive polynomial property:

→ A primitive polynomial generates all ~~negative~~ zero states exactly once ~~before~~ repeating.

ID = IT24612

→ Therefore, the sequence cycles through $2^m - 1$ distinct states.

③ conclusion:

→ maximum period of the keystreams: $2^m - 1$

→ note: If the polynomial is not primitive, the period is a factor of $2^m - 1$, but not maximal.

(25) In lattice-based cryptography, particularly using the learning with ERROR (LWE) problem and shortest vector problem (SVP), sign a message as follows:

(i) Explain the process of signing a message using an LWE-based signature scheme including the key generation, signing, and verification steps.

(ii) Given a message M, demonstrates how to sign the message using an LWE-based scheme with a public key PK and private key SK. Outline the steps of message signing and explain the role of the LWE problem in ensuring security.

ID = IT29612

Ans:
Lattice-based cryptography provide post-quantum secure digital signatures, once commonly used approach is based on the learning with error (LWE) problem, which is hard even for quantum computers

① process of signing a message using LWE-Based scheme.

step 1: Key Generation.

① choose a matrix $A \in Z_q^{n \times m}$ randomly

② sample a secret vector $S_k \in Z_q^m$ (private key)

③ compute the public key

$$P_K = A \cdot S_K + e \bmod q$$

where e is small error (sampled from a discrete Gaussian distribution)

step-2: signing a message M

① Hash the message to obtain a digest:
$h = H(m)$

② simple random lattice-vector y (short vector) for masking

ID = IT24612

③ Compute a signature σ using the secret key s_k and the public key p_k such that

$$\sigma = y + s_k \cdot f(h, y) \bmod q$$

where $f(h, y)$ is a deterministic function mapping message and randomness to a lattice challenge.

Step 3 : verification

① Use public key p_k to compute :

$$y v = A \cdot \sigma - f(h, y) \cdot p_k \bmod q$$

② check whether v is small (with the appropriate norm)

③ If yes → signature is valid ; otherwise → invalid

Security principle :

The difficulty of recovering s_k from $p_k = A \cdot s_k + e$ is equivalent to solving LWE, which is hard for classical and quantum computers.

(II) Example steps for signing a message.

Given,

→ message m

→ public key $p_k = A \cdot s_k + e$

→ private key s_k

step-by-step signing

① Hash the message :

$$h = H(m)$$

ID = IT24612

2. Sample a random vector y from a small Gaussian distribution.

3. Compute lattice-based signature:

$$\sigma = y + sk \cdot f(h, y) \bmod q$$

4. Send signature (σ) along with the message to the verifier.

Verification process:

(1) Compute $f(h, y)$ from the message and signature.

(2) Check

$$v = A \cdot \sigma - f(h, y) \cdot pk \bmod q$$

(3) If $\|v\|$ is small $\Rightarrow$ signature is valid, ensuring authenticity and integrity.