Q_1. Fermat's little theorem:

Theorem:

If p is a prime and $a \not\equiv 0 \mod p$, then

$$a^{p-1} \equiv 1 \mod p$$

Given: $a = 7, p = 13$

$$a^{p-1} = 7^{12} \mod 13$$

use successive squaring and modular exponentiation:

$$7^2 = 49 \mod 13 = 10$$

$$7^4 = 10^2 = 100 \mod 13 = 9$$

$$7^8 = 9^2 = 81 \mod 13 = 3$$

$$7^{12} = 7^4 \cdot 7^9 = 9 \cdot 3 = 27 \mod 13 = 1$$

Proved: $7^{12} \equiv 1 \mod 13$

Q_2. Euler's Totient Function:

Formula:

If $n = p_1^{k_1} \cdot p_2^{k_2} \cdots$, then

$$\phi(n) = n \prod \left(1 - \frac{1}{p_i}\right)$$

Mirza Zisun

$\phi(35) = 35 \cdot (1-\frac{1}{5})(1-\frac{1}{7}) = 35 \cdot \frac{4}{5} \cdot \frac{6}{7} = 24$

$\phi(45) = 45 \cdot (1-\frac{1}{3})(1-\frac{1}{5}) = 45 \cdot \frac{2}{3} \cdot \frac{4}{5} = 24$

$\phi(100) = 100 \cdot (1-\frac{1}{2})(1-\frac{1}{5}) = 100 \cdot \frac{1}{2} \cdot \frac{4}{5}) = 40$

If $\gcd(a, n) = 1$, then

$$a^{\phi(n)} \equiv 1 \bmod n$$

Q3. Chinese Remainden Theorem:

$x \equiv 2 \bmod 3$

$x \equiv 3 \bmod 4$

$x \equiv 4 \bmod 5$

Step 1: Convent to normal form

Notice:

$x \equiv -1 \bmod 3,4,5 \Rightarrow x \equiv -1 \bmod 60 \Rightarrow x \equiv 59 \bmod 60$

So, $x \equiv 59 \bmod 60$

To prove $x \equiv 11 \bmod 60$ is not correct,

Q4. Primitive Root modulo 17:

Primitive roots of prime $p = 17$ must satisfy

$g^k \not\equiv 1 \bmod 17$ unless $k = 16$

Try $g = 3$:

$3^1 = 3,\ 3^2 = 9,\ 3^4 = 13, \ldots, 3^{16} \equiv 1 \bmod 17$

$g = 3$ is a primitive root.

Q5. Carmichael Number check for 561

$561 = 3 \times 11 \times 17$ (all primes).

Check if:

$a^{561-1} \equiv 1 \bmod 561$ for all $\gcd(a, 561) = 1$

Q6. Discrete Logarithm:

Find x such that:

$3^x \equiv 13 \bmod 17$

Try powers of 3 mod 17:

$3^1 = 3$

$3^2 = 9$

$3^3 = 10$

$3^4 = 13$

Zisun

∴ Answer: $x = 4$.

Zisun

Q2. Role in Diffie-Hellman

→ Uses discrete logs for key exchange.

→ Hardness of computing $g^a \bmod p$ from g and g^a ensures security.

→ Enables secure shared secret generation

Q8. Cipher Comparison:

Cipher	Mechanism	Key space	Vulnerable to frequency
Substitution	Replace letters	Large	yes
Transposition	Rearrange letters	Medium	NO
Playfair	2-letter blocks	Larger than mono	Less

Plaintext : HELLO

Substitution : URYYB

Transposition : LOHEL

Playfair : Depends on matrix

Q9. Affine Ciphen :

Given :

$$E(x) = (5x+8) \bmod 26$$

Example :

D(3) → (5x3 +8) mod 26 = 23 → x

E(4) → (5x4+8) = 28 → 2 → C

P(15) → (5x15+8) mod 26 → 5 → F

T(19) → (5x19+8) mod 26 → 25 → Z

O(14) → (5x14+8) mod 26 → 0 → A

F(5) → (5x5+8) mod 26 → 7 → H

I(8) → (5x8+8) mod 26 → 22 → W

C(2) → (5x2+8) mod 26 → 18 → S

T(19) → (5x19+8) mod 26 → 25 → Z

M → (5x12+8) mod 26 → 16 → Q

B(1) → (5x1+8) mod 26 → 13 → N

S(18) → (5x18+8) mod 26 → 20 → U

T(19) → (5x19+8) mod 26 → 25 → Z

U(20) → (5x20+8) mod 26 → 4 → E

∴ Final Encrypted Text:

"XCFZAHWSZQNUZE"

Zisun

b) Decryption:

Decryption function:

$$D(y) = a^{-1}(y-b) \bmod 26$$

Where;

$a = 5$

$b = 8$

$a^{-1} = 21$ (since $5 \cdot 21 \equiv 1 \bmod 26$)

We now reverse each letters from Ciphertext "XCFZAHWSZQNUZE":

Letter	y	$D(y) = 21(y-8) \bmod 26$	Decrypted
X	23	$21 \times (23-8) = 21 \times 15 = 315 \to 3$	D
C	2	$21 \times (2-8) \bmod 26 = 4$	E
F	5	$21 \times (5-8) \bmod 26 = 15$	P
Z	25	$21 \times (25-8) \bmod 26 = 19$	T
A	0	$21 \times (0-8) \bmod 26 = 14$	O
H	7	$21 \times (7-8) \bmod 26 = 5$	F
W	22	$21 \times (22-8) \bmod 26 = 8$	I
S	18	$21 \times (18-8) \bmod 26 = 2$	C
Z	25	$21 \times (25-8) \bmod 26 = 19$	T
Q	16	$21 \times (16-8) \bmod 26 = 12$	M
N	13	$21 \times (13-8) \bmod 26 = 1$	B
U	20	$21 \times (20-8) \bmod 26 = 18$	S
Z	25	$21 \times (25-8) \bmod 26 = 19$	T
E	4	$21 \times (4-8) \bmod 26 = 20$	U

∴ Final Decrypted text: DEPTOFICTMBSTU

Zisun

Q10. Novel Cipher

Zisun

Encryption Process :

1. Key Generation Using PRNG

→ Substitution key : Shuffle the alphabet using a PRNG with fixed seed.

→ Permutation key : Generate a permutation pattern for a block of fixed size.

2. Substitution :

→ Replace each letter in the plaintext using shuffled alphabet.

3. Permutation :

→ Divide the substitution text into blocks

→ Rearrange characters in each block according to the permutation key.

Example :

Plaintext : "HELLO WORLD" → Remove spaces →
"HELLOWORLD"

Substitution : "ITSSGIVGIKSR"

Permutation : "SIIGITSGIRVK"

Ciphertext : "SIIGITSGIRVK"

Decryption process :

1. Reverse the permutation using the inverse of the key.

2. Reverse the substitution using the inverse shuffled alphabet.

Recovered plaintext : "HELLOWORLD"

Cryptanalysis (Weaknesses) :

→ Frequency analysis possible on substitution phase.

→ Fixed block size may leak Pattern.

→ Subsceptible to known plaintext attacks.

→ Brute-force possible for short messages