Q1. What type of cryptographic algorithm is RSA?

- Symmetric key algorithm
- Asymmetric key algorithm
- Hash function
- Stream cipher

Answer: Asymmetric key algorithm
Explanation: RSA is an asymmetric algorithm using a public and private key pair.

Q2. RSA is based on the difficulty of which mathematical problem?

- Integer factorization
- Discrete logarithm
- Elliptic curves
- Modular exponentiation

Answer: Integer factorization
Explanation: RSA's security relies on the difficulty of factoring large composite numbers.

Q3. In RSA, what do the public and private keys consist of?

- Public key: (e, n); Private key: (d, n)
- Public key: (d, n); Private key: (e, n)
- Public key: (p, q); Private key: (n, e)
- Public key: (d, p); Private key: (e, q)

Answer: Public key: (e, n); Private key: (d, n)
Explanation: Public key is the exponent e and modulus n; private key is exponent d and modulus n.

Q4. How is the modulus n computed in RSA?

- $n = p \times q$
- $n = p + q$
- $n = e \times d$
- $n = e + d$

Answer: $n = p \times q$
Explanation: The modulus n is the product of two large primes p and q.

Q5. What is the role of Euler's Totient function φ(n) in RSA?

- To help compute the private key d
- To encrypt the message
- To decrypt the message
- To generate random keys
  Answer: To help compute the private key d
  Explanation: $\varphi(n) = (p-1)(q-1)$ is used to find d such that $d \times e \equiv 1 \mod \varphi(n)$.

Q6. Which equation must hold true between e and d in RSA?

- $e \times d \equiv 1 \pmod{\varphi(n)}$
- $e + d = n$
- $e \times d = p \times q$
- $e - d = \varphi(n)$
  Answer: $e \times d \equiv 1 \pmod{\varphi(n)}$
  Explanation: The private exponent d is the modular inverse of e modulo $\varphi(n)$.

Q7. What is the typical choice for the public exponent e?

- 65537
- 3
- 17
- Any random large number
  Answer: 65537
  Explanation: 65537 is commonly chosen because it is a prime and efficient for encryption.

Q8. How does RSA encryption work?

- Ciphertext = plaintext^e mod n
- Ciphertext = plaintext^d mod n
- Ciphertext = (plaintext + e) mod n
- Ciphertext = (plaintext × d) mod n
  Answer: Ciphertext = plaintext^e mod n
  Explanation: Encryption raises the plaintext to the power of e modulo n.

Q9. How does RSA decryption work?

- Plaintext = ciphertext^d mod n
- Plaintext = ciphertext^e mod n

- Plaintext = (ciphertext - d) mod n
- Plaintext = (ciphertext × e) mod n

  Answer: Plaintext = ciphertext^d mod n

  Explanation: Decryption raises the ciphertext to the power of d modulo n.

Q10. Why must p and q be large prime numbers?

- To make factoring n difficult
- To speed up encryption
- To simplify key generation
- To reduce key size

  Answer: To make factoring n difficult

  Explanation: Large primes ensure n is hard to factor, securing RSA.

Q11. Which of the following is a major weakness if implemented incorrectly in RSA?

- Using small or predictable primes
- Choosing a large e
- Encrypting with d
- Using large modulus n

  Answer: Using small or predictable primes

  Explanation: Small or predictable primes make it easy to factor n and break RSA.

Q12. What is the purpose of padding schemes like OAEP in RSA?

- To prevent certain attacks and ensure semantic security
- To compress the message
- To speed up encryption
- To generate keys

  Answer: To prevent certain attacks and ensure semantic security

  Explanation: Padding helps defend against chosen plaintext and other attacks.

Q13. What does RSA digital signature provide?

- Authentication and integrity
- Encryption only
- Compression

- Key exchange
  Answer: Authentication and integrity
  Explanation: Signatures verify the sender's identity and that data hasn't been altered.

Q14. Which operation is used for signing in RSA?

- Signature = message^d mod n
- Signature = message^e mod n
- Signature = message × d mod n
- Signature = message × e mod n
  Answer: Signature = message^d mod n
  Explanation: The signer uses the private key exponent d to generate the signature.

Q15. What is a common key size for RSA today to ensure security?

- 2048 bits or higher
- 512 bits
- 1024 bits
- 256 bits
  Answer: 2048 bits or higher
  Explanation: 2048-bit keys are currently recommended; smaller keys are vulnerable.

Q16. Why is RSA considered slower than symmetric algorithms like AES?

- Because of complex mathematical operations on large numbers
- Because it uses longer keys
- Because it encrypts larger blocks
- Because it uses multiple rounds
  Answer: Because of complex mathematical operations on large numbers
  Explanation: RSA uses modular exponentiation on large numbers, which is computationally intensive.

Q17. What is hybrid encryption?

- Using RSA to encrypt a symmetric key, then AES to encrypt data
- Using RSA alone for all encryption

- Using AES to encrypt the RSA keys
- Using symmetric encryption only

  Answer: Using RSA to encrypt a symmetric key, then AES to encrypt data

  Explanation: Hybrid encryption combines strengths of asymmetric and symmetric cryptography.

Q18. Which of these is NOT an RSA usage?

- Encrypting large data files directly
- Key exchange
- Digital signatures
- Secure communication setup

  Answer: Encrypting large data files directly

  Explanation: RSA is usually not used for large data due to inefficiency; symmetric keys encrypt data instead.

Q19. What is the RSA assumption?

- Factoring n is hard
- Discrete log is hard
- Hashing is one-way
- Primes are easy to find

  Answer: Factoring n is hard

  Explanation: RSA's security assumes factoring the modulus n is computationally infeasible.