ETE

Here are common types of questions for each topic:

BT-7

1. **Cryptographic Hash Functions**
    - Define a cryptographic hash function.
    - What is collision resistance? Why is it important?
    - Example: Explain how SHA-256 is used in blockchain.
2. **Merkle Trees**
    - What is the purpose of a Merkle tree in blockchain?
    - How can you prove a specific transaction exists in a Merkle tree?
    - Example: Illustrate the structure of a Merkle tree for four transactions.
3. **Proof of Work**
    - What is the proof-of-work mechanism?
    - How does adjusting difficulty (D) influence mining?
    - Example: Describe the steps to verify a proof-of-work solution.
4. **Digital Signatures**
    - Explain the role of public and private keys in digital signatures.
    - What are the advantages of ECDSA over RSA?
    - Example: How does a digital signature ensure transaction authenticity?

1. Theory-Based Questions

- Define cryptographic hash functions and give an example.
- Explain the properties of collision-resistant hash functions.
- What is a Merkle tree? Why is it used in blockchain?

2. Application-Based Questions

- How does a Merkle tree help prove the validity of transactions?
- Describe the proof-of-work process used in Bitcoin.
- Explain how digital signatures prevent tampering in blockchain transactions.

---

3. Numerical/Diagrammatic Questions

- Construct a Merkle tree for given transactions.
- If a hash function outputs 256-bit values, what is the probability of collision after 2^{64} inputs?

- Given a hash puzzle H(x,y)<2256/DH(x, y) < 2^{256}/DH(x,y)<2256/D, calculate yyy for a given xxx and DDD.

---

4. Comparison/Analysis Questions

- Compare symmetric and asymmetric encryption.
- What are the benefits of ECDSA over RSA in blockchain?
- How do BLS signatures improve scalability in Ethereum 2.0?

---

5. Problem-Solving Questions

- Prove that a transaction belongs to a blockchain using Merkle proofs.
- Design a digital signature scheme for secure message verification.
- Analyze the efficiency of proof-of-work under increasing difficulty.

BT-8

1. Theory-Based Questions

- Explain the key differences between Bitcoin and Ethereum.
- What is a Merkle Patricia Trie? How is it used in Ethereum?
- Define smart contracts and explain their lifecycle in Ethereum.
- What are the types of Ethereum accounts?

2. Application-Based Questions

- How does a Merkle tree help prove the validity of transactions?
- Describe the proof-of-work process used in Bitcoin.
- Explain how digital signatures prevent tampering in blockchain transactions.

---

3. Numerical/Diagrammatic Questions

- Construct a Merkle tree for given transactions.

- If a hash function outputs 256-bit values, what is the probability of collision after 2^{64} inputs?
- Given a hash puzzle H(x,y)<2256/DH(x, y) < 2^{256}/DH(x,y)<2256/D, calculate yyy for a given xxx and DDD.

---

4. Comparison/Analysis Questions

- Compare symmetric and asymmetric encryption.
- What are the benefits of ECDSA over RSA in blockchain?
- How do BLS signatures improve scalability in Ethereum 2.0?

---

5. Problem-Solving Questions

- Prove that a transaction belongs to a blockchain using Merkle proofs.
- Design a digital signature scheme for secure message verification.
- Analyze the efficiency of proof-of-work under increasing difficulty.

1. Conceptual Questions

- What are the limitations of Bitcoin that Ethereum overcomes?
- Explain the role of the Ethereum Virtual Machine (EVM) in executing smart contracts.
- Describe the difference between externally owned accounts (EOAs) and contract accounts (CAs).

---

2. Practical Understanding

- Why does Ethereum use gas for transactions? How does it affect network security?
- Explain the significance of nonce in Ethereum transactions.
- How does the “base fee” in Ethereum prevent transaction spamming?

---

3. Advanced Topics

- What is the role of uncle blocks in Ethereum, and how are they rewarded?
- Discuss the use of ERC standards (ERC-20, ERC-721) in Ethereum applications.
- Explain the concept of composability in Ethereum smart contracts with examples.

---

4. Solidity and Smart Contracts

- Write a Solidity function to transfer tokens between two addresses.
- What are the advantages of using Solidity for smart contract development?
- Explain how NameCoin demonstrates the use of smart contracts for domain registration.

---

5. Blockchain Architecture

- How are transactions, states, and receipts stored in Ethereum?
- Explain the components of an Ethereum block body and their importance.
- How does Ethereum handle state transitions differently from Bitcoin?

---

6. Real-Life Applications

- Discuss how Ethereum enables DeFi applications like lending and stablecoins.
- What is the role of Merkle Patricia Trie in Ethereum storage and verification?
- How does Ethereum facilitate decentralized apps (DApps) through smart contracts

---

BT-8.1

Theory-based Questions:

- Definitions, concepts, and principles (e.g., "Define Ethereum Virtual Machine").
- Comparisons (e.g., "Compare blockchain and traditional databases").

Application-based Questions:

- Use case scenarios (e.g., "Explain the role of a miner in the Ethereum network").
- Implementation details (e.g., "How is a transaction executed in Ethereum?").

Problem-solving Questions:

- Calculations or design (e.g., "Design a smart contract for a voting system").
- Debugging code or analyzing blockchain transactions.

Diagram-based Questions:

- Explaining systems with diagrams (e.g., "Draw and explain the architecture of Ethereum").

Short Notes or Essays:

- Writing detailed explanations about specific topics (e.g., "Write a short note on GAS in Ethereum").

1. Objective-type Questions

- **MCQs**: Multiple choice questions for quick assessments (e.g., "Which consensus algorithm does Ethereum use by default?").
- **True/False**: For simple conceptual checks (e.g., "Blockchain is a centralized system: True or False").
- **Fill-in-the-blanks**: Testing key terms (e.g., "The smallest unit of Ether is called _____.").

---

2. Practical-based Questions

- **Code Analysis**: Examine provided code snippets and predict outputs or identify errors (e.g., "What will the output of this Solidity function be?").
- **Scenario-based Tasks**: Create or modify a system (e.g., "Write a smart contract for a crowdfunding platform").

---

3. Case Studies or Real-world Scenarios

- Students analyze a real-world application or problem and explain solutions (e.g., "Discuss how Ethereum can be used to improve supply chain management").
- Evaluate a situation and provide insights (e.g., "Analyze the pros and cons of Ethereum's switch to Proof of Stake in the context of energy efficiency").

---

4. Research-oriented Questions

- Questions encouraging exploration of recent advancements (e.g., "Explain how Layer 2 scaling solutions, such as Optimistic Rollups, enhance Ethereum's performance").
- Summary of research papers (e.g., "Discuss the core points of the Ethereum Yellow Paper").

---

5. Critical Thinking and Open-ended Questions

- Encourage debate or opinion-based answers (e.g., "Will blockchain disrupt traditional banking systems? Justify your answer").

- Hypothetical scenarios (e.g., "If Ethereum were to replace fiat currency, what changes would you expect in the global economy?").

---

6. Diagrammatic and Tabular Questions

- Require students to draw diagrams or flowcharts (e.g., "Illustrate the architecture of the Ethereum Virtual Machine (EVM)").
- Tabular comparisons (e.g., "Compare Ethereum and Bitcoin in terms of purpose, consensus, and scalability").

---

7. Interdisciplinary Questions

- Combining knowledge from different fields (e.g., "Discuss how blockchain principles can be applied in healthcare for maintaining patient data")

1. Multiple Choice Questions (MCQs)

1. **What type of state machine does Ethereum use?**
   - a) Rule-based
   - b) Transaction-based
   - c) Sequential
   - d) Temporal

   Answer: b) Transaction-based
2. **Which account type in Ethereum is controlled by a private key?**
   - a) Contract Account
   - b) Externally Owned Account (EOA)
   - c) Miner Account
   - d) System Account

   Answer: b) Externally Owned Account (EOA)
3. **What is the primary purpose of the GAS in Ethereum?**
   - a) Measure computation resources
   - b) Store transactions
   - c) Encrypt private keys
   - d) Validate accounts

   Answer: a) Measure computation resources

4. **Which consensus algorithm was Ethereum originally based on?**

- a) Proof of Stake
- b) Proof of Work
- c) Delegated Proof of Stake
- d) Byzantine Fault Tolerance
  Answer: b) Proof of Work

---

2. True/False Questions

1. Ethereum uses a decentralized peer-to-peer (P2P) network.
   Answer: True
2. Contract accounts in Ethereum are controlled by private keys.
   Answer: False
3. The Ethereum Virtual Machine (EVM) is stack-based.
   Answer: True
4. The nonce in a transaction is used to prevent double-spending.
   Answer: True

---

3. Fill-in-the-Blanks

1. Ethereum can be viewed as a _____-based state machine.
   Answer: transaction
2. The smallest unit of Ether is called _____.
   Answer: Wei
3. A blockchain can be described as a _____, decentralized database.
   Answer: globally shared
4. The two types of accounts in Ethereum are _____ and _____.
   Answer: Externally Owned Account (EOA), Contract Account

---

4. Match the Following

Column A	Column B
GAS	Measures computational costs

Externally Owned Account	Controlled by private key
Contract Account	Controlled by EVM code
SHA-3	Ethereum's hashing algorithm

---

5. Assertion and Reasoning

1. **Assertion (A)**: Ethereum is a blockchain platform that supports smart contracts.
   Reason (R): Smart contracts in Ethereum are written using the Solidity programming language.
   - a) Both A and R are true, and R is the correct explanation of A.
   - b) Both A and R are true, but R is not the correct explanation of A.
   - c) A is true, but R is false.
   - d) A is false, but R is true.
     Answer: a) Both A and R are true, and R is the correct explanation of A.

BT-8

Types of Questions

1. Objective Questions

1. **Multiple Choice Questions (MCQs)**:
   - What is the output size of SHA-256?
     a) 64 bytes
     b) 32 bytes
     c) 16 bytes
     d) 128 bytes
     Answer: b) 32 bytes
   - Which property ensures that it is "hard" to find a collision in a hash function?
     a) Compression
     b) Determinism
     c) Collision resistance

d) Preimage resistance
Answer: c) Collision resistance

2. **True/False**:
   - Digital signatures ensure both integrity and authenticity.
     Answer: True
   - RSA is the most commonly used signature algorithm in blockchains like Bitcoin and Ethereum.
     Answer: False (ECDSA is used).
3. **Fill-in-the-Blanks**:
   - The _____ algorithm is used for digital signatures in Bitcoin.
     Answer: Schnorr
   - Ethereum 2.0 employs _____ signatures for efficient validation.
     Answer: BLS (Boneh-Lynn-Shacham)

4. **Match the Following**:

Concept	Description
Merkle Tree	Data structure for verifying transactions
Collision Resistance	Property of cryptographic hash functions
Digital Signature	Ensures integrity and authenticity
Proof of Work	Consensus mechanism

---

2. Analytical or Problem-solving Questions

1. Explain how Merkle trees ensure data integrity in a blockchain.
2. Given a cryptographic hash function HHH, prove why collision resistance is essential for blockchain applications.

---

3. Diagram-based Questions

1. Draw and explain the structure of a Merkle tree for four transactions T1,T2,T3,T4T1, T2, T3, T4T1,T2,T3,T4.
2. Illustrate the process of generating and verifying a digital signature using public and private keys.

---

4. Scenario-based or Application Questions

1. Describe how digital signatures are used to validate transactions in blockchain systems.
2. In a Proof-of-Work system, explain why increasing the difficulty parameter DDD makes mining computationally expensive.

---

5. Research-oriented or Case Study Questions

1. Compare ECDSA, Schnorr, and BLS signature schemes in terms of scalability and efficiency for blockchain use cases.
2. Discuss the impact of quantum computing on cryptographic techniques like RSA and ECDSA.

Objectives for Questions

1. **Conceptual Understanding**:
    - Test foundational knowledge of cryptography (e.g., properties of hash functions, digital signatures).
2. **Application of Knowledge**:
    - Assess the ability to apply cryptographic concepts to real-world blockchain scenarios (e.g., use of Merkle trees in transaction verification).
3. **Critical Thinking**:
    - Evaluate analytical reasoning and problem-solving skills (e.g., impact of collision resistance failure on blockchain security).
4. **Technical Proficiency**:
    - Test practical knowledge, such as signature verification or building Merkle proofs.
5. **Research and Comparison**:
    - Explore advancements like post-quantum cryptography and how blockchain protocols adapt to emerging threats

Multiple Choice Questions (MCQs)

1. What is the primary advantage of a Merkle tree in blockchain systems?
    - a) Reduces computational complexity
    - b) Ensures faster block mining

    - c) Efficient verification of data integrity
    - d) Eliminates the need for digital signatures
      Answer: c) Efficient verification of data integrity
2. Which property of digital signatures ensures that a signed message cannot be forged?
    - a) Non-repudiation
    - b) Confidentiality
    - c) Scalability
    - d) Data compression
      Answer: a) Non-repudiation
3. Proof of Work (PoW) is best described as:
    - a) A cryptographic hash puzzle that is difficult to solve but easy to verify
    - b) A method to encrypt transactions in a blockchain
    - c) A consensus algorithm that uses voting
    - d) A way to store blocks efficiently
      Answer: a) A cryptographic hash puzzle that is difficult to solve but easy to verify
4. In Ethereum, which cryptographic algorithm is used for hashing?
    - a) SHA-1
    - b) SHA-256
    - c) Keccak-256
    - d) Blake2
      Answer: c) Keccak-256

True/False Questions

1. A Merkle root is the hash of all transactions in a block, organized in a binary tree.
   Answer: True
2. ECDSA is no longer used in Bitcoin after the switch to Proof-of-Stake.
   Answer: False
3. In Proof of Work, the difficulty parameter DDD controls how long it takes to mine a block.
   Answer: True
4. Public key cryptography requires the same key for encryption and decryption.
   Answer: False

Fill-in-the-Blanks

1. A _____ is a cryptographic function that maps input data of arbitrary size to a fixed size.
   Answer: hash function
2. The _____ algorithm ensures that data integrity is maintained in digital communications.
   Answer: digital signature

3. The _____ structure in blockchain allows verification of any transaction in logarithmic time.
   Answer: Merkle tree
4. _____ cryptography uses two keys: a public key and a private key.
   Answer: Asymmetric

Analytical Questions

1. **Explain the role of collision resistance in cryptographic hash functions. Provide an example of its application in blockchain systems.**
2. **Compare Proof of Work (PoW) and Proof of Stake (PoS) consensus mechanisms. Highlight their strengths and weaknesses.**

Scenario-based Questions

1. **Describe how a digital signature can be used to secure a financial transaction on the blockchain. Include steps for signature generation and verification.**
2. **You are given a Merkle root of a block. Explain how you would verify that a specific transaction is part of this block.**

Diagram-based Questions

1. **Illustrate the structure of a Merkle tree with five transactions (T1,T2,T3,T4,T5T1, T2, T3, T4, T5T1,T2,T3,T4,T5) and show the process to verify T3T3T3.**
2. **Draw and explain the process of a digital signature from key generation to signature verification.**

Case Study Questions

1. **How does Ethereum 2.0 use BLS signatures to improve scalability? Explain the advantages over traditional ECDSA.**
2. **Discuss the use of Proof of Work (PoW) in Bitcoin. Why is it energy-intensive, and what are the potential alternatives?**

Open-ended Questions

1. **What are the potential security risks of using older cryptographic algorithms, such as SHA-1, in modern blockchain systems?**
2. **In a post-quantum computing era, how should blockchain systems evolve to maintain security and integrity?**

BT-5

Objective Questions

Multiple Choice Questions (MCQs)

1. **What is the main principle of the Nakamoto Consensus?**
    - a) Proof of Stake and sharding
    - b) Leader-based block approval
    - c) Proof of Work and the longest chain rule
    - d) Byzantine Fault Tolerance

    Answer: c) Proof of Work and the longest chain rule
2. **Which factor determines the mining difficulty in Proof of Work?**
    - a) Number of transactions in a block
    - b) Number of leading zeros in the target hash
    - c) Hashing algorithm used
    - d) Blockchain size

    Answer: b) Number of leading zeros in the target hash
3. **What does the Nakamoto property of "chain quality" ensure?**
    - a) Consistency among all miners
    - b) Percentage of honestly mined blocks
    - c) Reduction in mining costs
    - d) Guaranteed block finality

    Answer: b) Percentage of honestly mined blocks
4. **What is the primary drawback of Proof of Work?**
    - a) Centralized decision-making
    - b) High energy consumption
    - c) Lack of security
    - d) Inability to prevent DoS attacks

    Answer: b) High energy consumption
5. **What is a "51% attack"?**
    - a) When a blockchain reaches 51 blocks
    - b) When an attacker controls more than half of the network's computational power
    - c) When a blockchain splits into multiple forks
    - d) When miners collaborate to reduce rewards

    Answer: b) When an attacker controls more than half of the network's computational power

---

True/False

1. The Nakamoto Consensus can tolerate up to 50% corruptions in the network.
   Answer: False (It tolerates up to less than 50%).
2. Proof of Work uses computational puzzles to ensure the integrity of blockchain transactions.
   Answer: True
3. Selfish mining requires an attacker to control more than 50% of the mining power.
   Answer: False (It can work with as little as 33%).
4. Orphan blocks are always included in the final blockchain.
   Answer: False

Fill-in-the-Blanks

1. Nakamoto Consensus combines _____ and the _____ rule to secure blockchain networks.
   Answer: Proof of Work, longest chain
2. In Proof of Work, the computational puzzle difficulty is adjusted based on _____.
   Answer: network load and mining rate
3. The _____ property ensures that honest miners contribute the majority of blocks in a blockchain.
   Answer: chain quality
4. A _____ attack occurs when an attacker keeps a private fork to invalidate honest miners' blocks.
   Answer: selfish mining

Analytical or Application Questions

1. **Explain the relationship between mining difficulty and blockchain security in Proof of Work systems.**
2. **Analyze how a 51% attack can affect the blockchain network. Provide examples from real-world incidents.**
3. **Why does Nakamoto Consensus require blocks to have a delay between proposals? How does it maintain network stability?**

Diagram-based Questions

1. **Draw and explain the Proof of Work mining process, including the role of the nonce and the target hash.**

2. **Illustrate how a selfish miner operates to gain an advantage over honest miners in a blockchain network.**

Scenario-based Questions

1. **A blockchain network is experiencing frequent forks. How can Nakamoto Consensus properties, like chain quality and consistency, help reduce such issues?**
2. **In a Proof of Work system, explain the impact of adjusting the block time from 10 minutes to 5 minutes on network security and efficiency.**

Research and Case Study Questions

1. **Discuss the energy consumption issues associated with Proof of Work and compare it with Proof of Stake as an alternative.**
2. **Evaluate the Nakamoto Consensus in handling network delays. What are its limitations, and how can they be addressed?**

Open-ended Questions

1. **Can Nakamoto Consensus be scaled for high-throughput applications like DeFi? Discuss its challenges and potential solutions.**
2. **What would happen if miners collaborated to form a centralized mining pool controlling 60% of the network? Analyze its implications on decentralization**

Objective Questions

Multiple Choice Questions (MCQs)

1. **What is the key property of Nakamoto Consensus that ensures blocks are added to the longest chain?**
    - a) Fork resolution
    - b) Leader election
    - c) Mining rewards
    - d) Proof of Work

    Answer: d) Proof of Work
2. **What happens to orphaned blocks in a blockchain network?**
    - a) They are included in the chain later.
    - b) They remain unused but visible.

- c) They are permanently discarded.
- d) They are merged into the longest chain.
  Answer: b) They remain unused but visible.

3. **Which of the following is an example of a "double-spending attack"?**
   - a) Deliberate forking to censor transactions
   - b) Spending the same cryptocurrency on two different transactions
   - c) Selfish mining to invalidate honest blocks
   - d) Changing the hash function used by the network
     Answer: b) Spending the same cryptocurrency on two different transactions
4. **What does "chain growth" in Nakamoto Consensus signify?**
   - a) The blockchain can accommodate more nodes.
   - b) Honest blocks are added at a steady rate.
   - c) Network delays are minimized.
   - d) Forks are automatically resolved.
     Answer: b) Honest blocks are added at a steady rate.
5. **What determines the difficulty of mining in a Proof of Work system?**
   - a) Size of transactions in the block
   - b) Total number of miners in the network
   - c) Target hash and network load
   - d) Blockchain size
     Answer: c) Target hash and network load

True/False Questions

1. The Nakamoto Consensus guarantees finality after kkk blocks.
   Answer: True
2. A selfish mining attack requires the attacker to control at least 51% of the network's computational power.
   Answer: False (It can work with as little as 33%).
3. Mining pools reduce the variance in mining rewards for individual miners.
   Answer: True
4. Consistency in Nakamoto Consensus means all nodes must have identical copies of the blockchain at all times.
   Answer: False (Consistency tolerates minor delays for kkk blocks).

Fill-in-the-Blanks

1. _____ ensures that only one valid block is added to the blockchain at a time.
   Answer: Proof of Work

2. A _____ is a block mined at the same time as another block but not included in the longest chain.
   Answer: fork (or orphan block)
3. _____ mining is an attack strategy where a miner keeps a private chain to maximize rewards.
   Answer: Selfish
4. The Nakamoto Consensus is designed to tolerate up to _____ corruptions in the network.
   Answer: less than 50%

Analytical Questions

1. **Explain why a "51% attack" is considered the most severe threat to Proof of Work blockchains. Discuss the implications for decentralization.**
2. **Analyze the relationship between mining difficulty and block time in a Proof of Work blockchain. How does adjusting the difficulty impact network performance and security?**

Diagram-based Questions

1. **Illustrate the structure of a Proof of Work block, including components like nonce, timestamp, and Merkle root. Explain how the hash is computed.**
2. **Draw and explain the effects of a selfish mining attack on the blockchain. Highlight how the attacker's chain can overtake the honest chain.**

Scenario-based Questions

1. **A blockchain network with a 10-minute block time is experiencing high network delays. Discuss how this affects the Nakamoto Consensus properties, including consistency and chain growth. Suggest mitigation strategies.**
2. **Suppose an adversary controls 40% of the network's computational power. How likely is it for the adversary to successfully execute a double-spend attack? Use Nakamoto Consensus properties to justify your answer.**

Research and Case Study Questions

1. **Evaluate the energy consumption issues associated with Proof of Work. Compare it to Proof of Stake and analyze whether PoS can fully address PoW's inefficiencies.**
2. **Study the impact of mining pools on the decentralization of Bitcoin. How do they influence the Nakamoto Consensus's robustness against attacks?**

Open-ended Questions

1. **Can Nakamoto Consensus adapt to faster block times without sacrificing security? Discuss the trade-offs involved.**
2. **If the majority of miners shifted to renewable energy sources, would Proof of Work become a sustainable consensus mechanism? Why or why not**

BT-6

1. Definitions and Concepts

- **Question: Define the following terms:**
  a) Proof of Work (PoW)
  b) Byzantine Fault Tolerance (BFT)
  c) Proof of Stake (PoS)
- **Question: What is the Byzantine Generals Problem, and how is it solved in blockchain systems?**

---

2. Consensus Mechanisms

- **Question: Explain the components of a blockchain consensus protocol.**
- **Question: Compare Proof of Work (PoW) and Proof of Stake (PoS) mechanisms.**

---

3. Types of Consensus Algorithms

- **Question: Briefly describe the following consensus algorithms:**
  a) Proof of Capacity (PoC)
  b) Proof of Elapsed Time (PoET)
  c) Proof of History (PoH)
- **Question: What is the role of incentives in blockchain consensus protocols?**

---

4. Detailed Protocol Questions

- **Question: Describe the Chain-based PoS mechanism. How does it differ from Committee-based PoS?**
- **Question: What are the key features of Ouroboros and its variations (Ouroboros Praos)?**

---

5. Security and Vulnerabilities

- **Question: List and explain at least three vulnerabilities associated with PoS mechanisms.**
- **Question: How does the Dolev-Strong protocol ensure consensus in distributed systems?**

---

6. Practical Applications and Examples

- **Question: Discuss the role of Tendermint in blockchain networks.**
- **Question: Provide examples of blockchain platforms that use Delegated Proof of Stake (DPoS).**

---

7. Analysis and Comparison

- **Question: Compare the energy efficiency of PoW and PoS.**
- **Question: Analyze how consensus protocols ensure fault tolerance in distributed systems**

Objective-Type Questions

1. **Which of the following is NOT a consensus mechanism?**
   a) Proof of Stake (PoS)
   b) Proof of Work (PoW)
   c) Proof of Elapsed Time (PoET)
   d) Byzantine Generals Problem (BGP)
   Answer: d) Byzantine Generals Problem (BGP)
2. **What is the primary goal of a consensus mechanism in blockchain?**
   a) Increasing transaction speed
   b) Ensuring agreement on the state of the distributed ledger
   c) Reducing computational costs

d) Preventing all forms of network errors
Answer: b) Ensuring agreement on the state of the distributed ledger

3. **Which protocol is specifically designed to handle asynchronous networks in blockchain?**
   a) Ouroboros Praos
   b) Proof of Burn (PoB)
   c) Tendermint
   d) Proof of History (PoH)
   Answer: a) Ouroboros Praos
4. **In the Nakamoto Consensus Protocol, which rule ensures that the longest chain is accepted?**
   a) Validation Rule
   b) Gossiping Rule
   c) Longest-Chain Rule
   d) Finalization Rule
   Answer: c) Longest-Chain Rule
5. **What happens if a validator acts dishonestly in a Proof of Stake (PoS) system?**
   a) They are removed from the network permanently.
   b) They lose their staked assets.
   c) They face legal action.
   d) They can propose only one block per epoch.
   Answer: b) They lose their staked assets.

Fill-in-the-Blanks

1. **In blockchain, the Byzantine Generals Problem illustrates the difficulty in reaching a single ___ in a decentralized network.**
   Answer: truth
2. **Proof of Work (PoW) requires miners to solve a ___ to generate a block.**
   Answer: cryptographic puzzle
3. **In Proof of Stake (PoS), the chance to propose a block is proportional to the ___ of the participant.**
   Answer: stake value
4. **The Ouroboros protocol uses a ___ process to generate the block leader sequence.**
   Answer: Multi-Party Computation (MPC)
5. **___ is a PoS-based blockchain platform that mitigates synchronization issues with features like empty slots and VRF.**
   Answer: Ouroboros Praos

BT-10

Section 1: Short Answer/Descriptive Questions

1. Explain the differences between hot wallets and cold wallets with examples.
2. Describe the process of Simplified Payment Verification (SPV) and its importance in blockchain wallets.
3. How does a blockchain wallet differ from traditional banking systems in terms of security and transaction handling?
4. What are the key features of a blockchain wallet? Provide examples of popular wallets.
5. Discuss the process of creating and using a hardware wallet for managing cryptocurrency.
6. Explain how the recovery phrase (24 words) is generated and used in hardware wallets. Why is it essential to protect it?
7. Describe how Merkle trees are used in blockchain wallets for transaction verification.
8. What is the role of a private key in a blockchain wallet, and how does its compromise affect the wallet owner?
9. Differentiate between software, hardware, and paper wallets with examples.
10. How does a blockchain wallet ensure privacy and identity protection during transactions?

Section 2: Multiple Choice Questions (MCQs)

1. **What type of wallet is best for day-to-day transactions?**
   a) Hardware wallet
   b) Paper wallet
   c) Hot wallet
   d) Cold wallet
   Answer: c) Hot wallet
2. **Which of the following is NOT a type of blockchain wallet?**
   a) Brain wallet
   b) Cloud wallet
   c) Debit card wallet
   d) Paper wallet
   Answer: c) Debit card wallet
3. **Which cryptographic concept is used to generate wallet addresses?**
   a) SHA-256
   b) RSA
   c) Base58 encoding
   d) HMAC
   Answer: c) Base58 encoding

4. **What is stored in a blockchain wallet?**
   a) Cryptocurrency coins
   b) Public and private keys
   c) Both (a) and (b)
   d) Neither (a) nor (b)
   Answer: b) Public and private keys
5. **What is the major drawback of a hot wallet?**
   a) Inconvenience
   b) Online vulnerability to hacking
   c) High costs
   d) Lack of transaction history
   Answer: b) Online vulnerability to hacking

Section 3: True/False Questions

1. A blockchain wallet stores cryptocurrencies. **(False)**
2. A private key in a blockchain wallet is similar to a password. **(True)**
3. SPV requires downloading the entire blockchain to verify transactions. **(False)**
4. Hardware wallets are more secure than online wallets. **(True)**
5. A Merkle proof uses a tree structure to validate transactions. **(True)**

Section 4: Fill in the Blanks

1. A wallet address is generated from a _____. (**Public key**)
2. _____ wallets are offline wallets that provide higher security by signing transactions offline. (**Cold**)
3. The recovery phrase used in hardware wallets usually consists of _____ words. (**24**)
4. In blockchain wallets, the private key is used to _____ transactions. (**sign**)
5. Blockchain wallets maintain _____ and _____ to ensure secure cryptocurrency transactions. (**privacy**, **identity**)

Section 5: Problem-Solving Questions

1. A user loses their hardware wallet. Explain the steps they should follow to recover their crypto assets securely.
2. Design a comparison chart of different wallet types based on security, ease of use, and cost.
3. Explain how a blockchain wallet verifies an incoming payment using Merkle proofs.

Section 1: Conceptual Questions

1. Why is the phrase "Not your keys, not your coins" significant in cryptocurrency management?
2. Discuss the differences between custodial and non-custodial wallets. Provide examples of each.
3. What are the risks of using a brain wallet, and why is it considered a bad idea?
4. How do hardware wallets ensure security against computer viruses and malware?
5. Explain how hybrid wallets work and their advantages over traditional hot and cold wallets.
6. What are threshold signatures, and how are they used in non-custodial cloud wallets?

Section 2: Real-World Scenarios and Analysis

1. A user accidentally shares their private key online. What are the immediate steps they should take to secure their funds?
2. Compare the usability and security of mobile wallets versus desktop wallets for everyday transactions.
3. If a wallet address is compromised, can the funds associated with that address be secured? Explain.
4. Discuss how blockchain wallets can help reduce transaction delays compared to traditional banking systems.
5. Analyze the impact of losing the recovery phrase of a hardware wallet. What measures can users take to prevent this?

Section 3: Multiple Choice Questions (MCQs)

1. **What is the primary purpose of a blockchain wallet?**
   a) To store cryptocurrency coins
   b) To manage and secure cryptographic keys
   c) To directly validate transactions on the blockchain
   d) To create new cryptocurrencies
   Answer: b) To manage and secure cryptographic keys
2. **What is the primary feature of a cold wallet?**
   a) It is connected to the internet
   b) It enables high-speed transactions
   c) It stores keys offline for enhanced security
   d) It is used for mining cryptocurrencies
   Answer: c) It stores keys offline for enhanced security
3. **Which of the following wallets would you recommend for a long-term investor prioritizing security?**

a) Mobile wallet
b) Hardware wallet
c) Web wallet
d) Brain wallet
Answer: b) Hardware wallet

4. **What technology does a blockchain wallet use to secure transactions?**
   a) Symmetric encryption
   b) Public-key cryptography
   c) Tokenization
   d) Blockchain forks
   Answer: b) Public-key cryptography
5. **Which of the following is true about SPV (Simplified Payment Verification)?**
   a) It requires downloading the entire blockchain.
   b) It uses Merkle proofs to verify transactions.
   c) It is used exclusively by hardware wallets.
   d) It eliminates the need for block headers.
   Answer: b) It uses Merkle proofs to verify transactions.

Section 4: True/False Questions

1. A blockchain wallet's public key can be shared with anyone to receive funds. **(True)**
2. A paper wallet is a type of hot wallet. **(False)**
3. Hardware wallets can be used to manage multiple cryptocurrencies securely. **(True)**
4. Transactions in blockchain wallets are cryptographically signed for security. **(True)**
5. The recovery phrase of a hardware wallet is stored on the blockchain. **(False)**

Section 5: Fill in the Blanks

1. The _____ wallet type provides the highest level of security by keeping private keys offline. (**cold**)
2. A _____ tree is used in SPV for verifying transactions without downloading the entire blockchain. (**Merkle**)
3. The process of generating unlinkable addresses for privacy is achieved using _____ wallets. (**HD wallets**)
4. If a private key is lost, the associated funds are _____. (**irrecoverable**)
5. Non-custodial wallets rely on the user's ability to manage their _____. (**private keys**)

Section 6: Problem-Solving/Programming Questions

1. Write a pseudocode to illustrate how a blockchain wallet generates a new address using a private key.
2. Design a system architecture for integrating a hardware wallet with a mobile app for cryptocurrency management.
3. Develop a flowchart explaining the process of verifying a transaction using SPV and Merkle proofs.
4. Discuss the potential risks associated with using cloud wallets and propose solutions to mitigate these risks.
5. Imagine a scenario where a blockchain wallet provider is hacked. How can the affected users recover or protect their funds?

BT-11

1. Understanding Consensus Mechanisms

- **Purpose**: Allow distributed networks to agree on a single version of truth (blockchain state) without central authority.
- **Types**:
    - **Permissionless**: Open participation (e.g., Bitcoin, Ethereum).
    - **Permissioned**: Restricted access to authenticated nodes (e.g., Apla, Ethereum Private).

---

2. Key Consensus Mechanisms

Proof of Authority (PoA):

- **Key Points**:
    - Replaces PoW, relies on validators' identities and reputation.
    - Validators go through certification; documents are publicly available.
    - Suitable for high performance and fault tolerance, especially in private chains.
    - Applications: Ethereum testnets (Rinkeby, Kovan), VeChain, POA Network.
- **Security**:
    - Relies on trust in small validator groups.
    - Vulnerable to 51% attacks on validator nodes.

Proof of Elapsed Time (PoET):

- **Key Points**:
    - Developed by Intel for Hyperledger Sawtooth.

    - Simulates time delays using **Trusted Execution Environment (TEE)**.
    - TEE ensures secure random back-off before block generation.
- **Challenges**:
    - Vulnerable to TEE attacks (e.g., Foreshadow, speculative execution).
    - Single vendor risk (Intel SGX).

Proof of TEE-Stake (PoTS):

- **Key Points**:
    - Combines TEE with staking mechanisms.
    - Resists Sybil attacks and offers better security than PoET.
    - Suitable for permissionless blockchain networks.

Proof of Retrievability (PoR):

- **Key Points**:
    - Focuses on storage resources instead of computational power.
    - Ensures file storage integrity (used by Permacoin).
    - Encourages decentralized and meaningful storage mining.

---

3. Specialized Consensus Mechanisms

Ripple Consensus Protocol/Algorithm (RCPA):

- **Key Points**:
    - Uses Unique Node Lists (UNLs) to define trusted nodes.
    - High performance, but limited decentralization due to strict trust requirements.

BlockDAG-Based Protocols:

- **Key Points**:
    - DAG structures (Directed Acyclic Graphs) allow for multiple parent blocks, increasing transaction throughput.
    - **SPECTRE** and **PHANTOM** address parent-selection mechanisms to prevent double-spending.

---

4. Key Challenges in Consensus Protocols

1. **Scalability**:
    - Blockchains struggle with low transaction rates compared to centralized systems like VISA.
    - Solutions like DAGs and sidechains are being explored.

2. **Security Risks**:
    - Vulnerabilities in TEE hardware.
    - Risk of Sybil attacks in protocols without robust identity checks.
3. **Energy Efficiency**:
    - Protocols like PoET and PoR aim to reduce the high energy costs of PoW.

---

5. Exam Focus

- Compare and contrast major consensus mechanisms (PoW, PoS, PoA, etc.).
- Discuss scalability solutions and challenges in blockchain adoption.
- Analyze security concerns specific to each protocol.
- Explain real-world applications of DAG and permissioned/permissionless systems.

Multiple Choice Questions (MCQs):

1. **Which consensus mechanism is primarily used by Bitcoin?**
   a) Proof of Stake (PoS)
   b) Proof of Work (PoW)
   c) Delegated Proof of Stake (DPoS)
   d) Practical Byzantine Fault Tolerance (PBFT)
   Answer: b
2. **What is the main goal of a consensus mechanism in blockchain networks?**
   a) To enhance the graphical user interface
   b) To ensure all participants agree on the data
   c) To distribute tokens among users
   d) To limit the transaction speed
   Answer: b
3. **Which of the following best describes the Proof of Stake (PoS) mechanism?**
   a) Miners compete to solve cryptographic puzzles.
   b) Participants create blocks based on their staked cryptocurrency.
   c) Validators are chosen randomly without any criteria.
   d) Consensus is achieved through external auditors.
   Answer: b
4. **In Delegated Proof of Stake (DPoS), who validates the transactions?**
   a) All participants in the network
   b) A randomly chosen validator
   c) Elected delegates
   d) Only the network administrator
   Answer: c

True/False Questions:

1. **Proof of Work (PoW) is an energy-efficient consensus mechanism.**
   Answer: False
2. **Consensus mechanisms are designed to prevent double-spending in blockchain networks.**
   Answer: True
3. **Proof of Stake (PoS) requires participants to solve complex cryptographic puzzles.**
   Answer: False
4. **Byzantine Fault Tolerance can handle up to one-third of malicious participants in a network.**
   Answer: True

Fill in the Blanks:

1. In Proof of Work (PoW), miners compete to solve _____ to validate transactions.
   Answer: cryptographic puzzles
2. Proof of Stake (PoS) selects validators based on the _____ of cryptocurrency they own and are willing to lock up.
   Answer: amount
3. The consensus mechanism used by Ethereum 2.0 is _____.
   Answer: Proof of Stake (PoS)
4. _____ is a consensus mechanism where a small number of participants are selected to validate transactions, reducing the network's energy consumption.
   Answer: Delegated Proof of Stake (DPoS)

Consensus Protocols Overview

1. **General Features**:
   - Many protocols are built on established schemes like Proof of Work (PoW), Proof of Stake (PoS), and Byzantine Fault Tolerance (BFT).
   - Others focus on specific aspects of blockchain consensus to address application-specific needs.
   - Consensus mechanisms empower nodes to validate transactions and update the blockchain network.
2. **Types of Blockchains**:
   - **Permissionless Blockchains** (e.g., Bitcoin, Ethereum): Open networks where anyone can join.
   - **Permissioned Blockchains** (e.g., Apla, Ethereum Private): Pre-authenticated nodes enable faster transaction rates.

Notable Consensus Mechanisms

1. **Proof of Authority (PoA)**:
    - Identity-based staking mechanism.
    - Validators are trusted entities that pass a mandatory certification process.
    - Features:
        - No mining; trusted validators maintain the network.
        - High transaction rates and scalability.
        - Used by platforms like VeChain, Palm Network, and Xodex.
    - Security:
        - Small, trusted groups mitigate 51% attacks.
        - Validators are publicly monitored to prevent misconduct.
2. **Proof of Elapsed Time (PoET)**:
    - Developed by Intel for permissioned networks (e.g., Hyperledger Sawtooth).
    - Simulates mining time using a **Trusted Execution Environment (TEE)**.
    - Features:
        - Reduces energy inefficiency of PoW.
        - Random back-off mechanism inside a TEE ensures fairness.
    - Security:
        - Vulnerable to hardware-level attacks on TEE (e.g., Foreshadow).
        - Best suited for authenticated networks.
3. **Proof of Retrievability (PoR)**:
    - Storage-based consensus for systems like **Permacoin**.
    - Features:
        - Proves secure file storage and retrievability.
        - Economical: Low energy consumption and reusable storage.
    - Security:
        - Resistant to ASIC dominance.
        - Requires randomness in storage distribution to prevent collusion.
4. **Ripple Consensus Protocol/Algorithm (RCPA)**:
    - Used in Ripple's payment network.
    - Features:
        - Validator nodes manage transactions using Unique Node Lists (UNL).
        - Trades fault tolerance for performance.
    - Security:
        - Fault tolerance capped at 1/5.
        - Requires high synchrony among nodes.
5. **BlockDAG-Based Consensus Protocols**:
    - Non-linear ledger structures for improved scalability.
    - Two types:
        - **Block-based DAG (BlockDAG)**: Flexible block referencing.
        - **Transaction-based DAG (TxDAG)**: Focuses on individual transactions.
    - Examples:

        - **SPECTRE**: Proposes parent-selection strategies to manage transaction flow.
        - **PHANTOM**: Addresses scalability and security challenges like double-spending.

Emerging Innovations

1. **Proof of TEE-Stake (PoTS)**:
    - Combines TEE and staking for enhanced security.
    - Features:
        - Suitable for permissionless blockchains.
        - Prevents grinding and Sybil attacks.
    - Challenges:
        - Still depends on single TEE hardware vendors, posing centralization risks.
2. **Sidechains and Directed Acyclic Graphs (DAG)**:
    - Solutions to enhance scalability.
    - BlockDAG protocols adapt PoW principles to allow flexible block ordering.
    - DAG innovations aim to process transactions in parallel, overcoming linear blockchain limitations.

Key Takeaways

- Blockchain platforms are innovating to address scalability, security, and decentralization challenges.
- New mechanisms like PoA, PoET, and BlockDAG emphasize performance, energy efficiency, and flexibility.
- Security remains a concern, especially for hardware-dependent protocols like PoET.

Multiple-Choice Questions (MCQs)

1. Which of the following is a key feature of Proof of Authority (PoA) as a consensus mechanism?
   A. Mining-based validation of transactions
   B. Stake of monetary tokens
   C. Validator stakes identity and reputation
   D. Focus on computational power
   Answer: C
2. Which consensus protocol simulates time-consuming processes of PoW using a trusted execution environment (TEE)?
   A. Proof of Stake (PoS)
   B. Proof of Elapsed Time (PoET)

C. Proof of Work (PoW)
D. Ripple Consensus Protocol Algorithm (RCPA)
Answer: B

3. What is the Byzantine Fault Tolerance threshold for Hyperledger Sawtooth's PoET?
   A. 1/3
   B. 50%
   C. 1/5
   D. 2/3
   Answer: B
4. What is the primary advantage of Proof of Retrievability (PoR) over Proof of Work (PoW)?
   A. Higher energy consumption
   B. Dependence on mining hardware
   C. Re-purposing for meaningful storage tasks
   D. Centralized control of transactions
   Answer: C

True or False

1. Ripple Consensus Protocol Algorithm (RCPA) allows up to 1/3 of faulty nodes in the UNL clique.
   False
2. In Proof of Authority (PoA), validators stake monetary tokens to participate in consensus.
   False
3. Block-DAG consensus protocols are designed to limit transaction throughput by restricting block size.
   False
4. Proof of Retrievability (PoR) depends on storage resources rather than computational power for earning cryptocurrency.
   True

Fill in the Blanks

1. In Proof of Authority (PoA), validators stake their _____ instead of monetary tokens.
   Answer: identity and reputation
2. Proof of Elapsed Time (PoET) relies on a trusted execution environment, such as Intel's _____ technology.
   Answer: Software Guard Extension (SGX)
3. A key feature of Proof of Retrievability (PoR) is allowing a file owner to verify that its _____ are securely stored and retrievable.
   Answer: files or file fragments

4. In Hyperledger Sawtooth's PoET, each node generates a certificate of back-off completion within the _____.
   Answer: enclave
5. The Ripple Consensus Protocol Algorithm (RCPA) uses a _____ node list (UNL) to identify trusted nodes for message exchanges.
   Answer: unique

Long Questions

1. **Describe the role of a trusted execution environment (TEE) in Proof of Elapsed Time (PoET).**
   - Mention its purpose, integration with the consensus process, and how it ensures trust and security.
2. **Compare and contrast Proof of Authority (PoA) and Proof of Stake (PoS) mechanisms.**
   - Highlight differences in validation criteria, scalability, and performance.
3. **Explain the scalability challenges of traditional blockchain and how Block-DAG protocols aim to address them.**
   - Discuss differences in structure and the impact on transaction throughput.

Short Answer Questions

1. What is the main advantage of using Proof of Authority (PoA) over Proof of Work (PoW)?
   Answer: PoA uses validators' identity and reputation instead of computational power, making it more energy-efficient and scalable.
2. Why does Proof of Elapsed Time (PoET) rely on a Trusted Execution Environment (TEE)?
   Answer: To ensure the fairness and randomness of the back-off time without external manipulation or influence.
3. Define Byzantine Fault Tolerance and its relevance in consensus algorithms.
   Answer: Byzantine Fault Tolerance is the ability of a system to reach consensus even if some nodes are malicious or faulty. It ensures trust in distributed systems like blockchain.

Case-Based Questions

Scenario:
A blockchain-based voting system is being developed for a local election. The developers are considering using Proof of Authority (PoA) for the system.

1. **Question:** Explain why Proof of Authority might be a suitable choice for this voting system.
   Answer: PoA is suitable because it relies on trusted validators, ensuring security and trustworthiness without requiring energy-intensive computations, ideal for a controlled environment like voting.
2. **Question:** If one of the validators in this PoA system is compromised, how might it affect the system, and what mitigation steps could be taken?
   Answer: A compromised validator might tamper with votes or disrupt consensus. Mitigation could involve quickly identifying and replacing the validator while maintaining a robust monitoring system.

Matching Questions

Match the following consensus protocols with their key features:

Consensus Protocol	Key Feature
Proof of Authority (PoA)	A. Validators stake identity
Proof of Work (PoW)	B. Mining for block validation
Proof of Stake (PoS)	C. Monetary tokens are staked
Proof of Elapsed Time (PoET)	D. Use of Trusted Execution Environment
Proof of Retrievability (PoR)	E. Focus on data storage

Answer:
1 → A, 2 → B, 3 → C, 4 → D, 5 → E

Assertion-Reason Questions

1. **Assertion:** Proof of Work (PoW) is less energy-efficient compared to Proof of Stake (PoS).
   Reason: PoW requires intensive computational effort to validate transactions.
   A. Both Assertion and Reason are true, and the Reason explains the Assertion.
   B. Both Assertion and Reason are true, but the Reason does not explain the Assertion.
   C. The Assertion is true, but the Reason is false.
   D. The Assertion is false, but the Reason is true.
   Answer: A
2. **Assertion:** Block-DAG structures have higher transaction throughput than traditional blockchains.
   Reason: Block-DAG protocols use a directed acyclic graph to allow parallel processing of transactions.
   Answer: A

Diagram-Based Question

Question:
Draw a diagram to represent the differences between a blockchain and a Block-DAG structure. Highlight how Block-DAG enables higher scalability.

Problem-Solving Question

Question:
A network is using Proof of Stake (PoS) for consensus. If a validator with a 30% stake fails to validate transactions correctly, how will this impact the system's functionality? Suggest solutions to mitigate such risks.
Answer:

- Impact: The validator’s failure could slow down the consensus process and undermine trust.
- Solutions: Implement slashing mechanisms to penalize faulty validators and decentralize stakes to reduce reliance on a single entity.