Topic 3.2: Blockchain Mechanics

Consensus, Blocks, and the Trilemma

Joerg Osterrieder

Digital Finance

2025

Learning Objectives

By the end of this topic, you will be able to:

1. **Define** what a blockchain is and explain its key properties (distributed, append-only, consensus-based)
2. **Describe** the structure of a block and how blocks are linked through cryptographic hashes
3. **Compare** Proof of Work (PoW) and Proof of Stake (PoS) consensus mechanisms
4. **Explain** the blockchain trilemma and why it creates fundamental design tradeoffs
5. **Evaluate** Layer 1 and Layer 2 scaling solutions

Core Question

How do thousands of strangers agree on a single version of the truth without any central authority?

Prerequisites: Cryptography Concepts You'll Need

From Topic 3.1 – Quick Recap:

Hash Functions

- Input → Fixed-size output (256 bits)
- **Deterministic:** Same input = same output
- **One-way:** Cannot reverse to find input
- **Avalanche effect:** Tiny change = completely different output

Example:
`"Hello"` → `185f8db3...`
`"Hello!"` → `334d016f...`

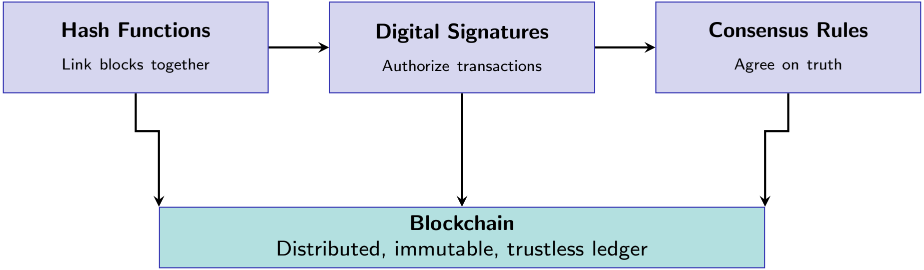
Digital Signatures

- Private key signs transactions
- Public key verifies signatures
- **Authentication:** Proves identity
- **Non-repudiation:** Cannot deny signing

Key insight:
Only the person with the private key can create a valid signature.

If these concepts are unfamiliar, review Topic 3.1 before continuing

How cryptographic primitives enable blockchain:



Today's focus: How these pieces fit together to create a system that works without trusted intermediaries.

What Is a Blockchain?

"A blockchain is a distributed ledger that is append-only, cryptographically linked, and maintained by consensus."

Distributed

- No central server
- Thousands of copies
- No single point of failure

Append-Only

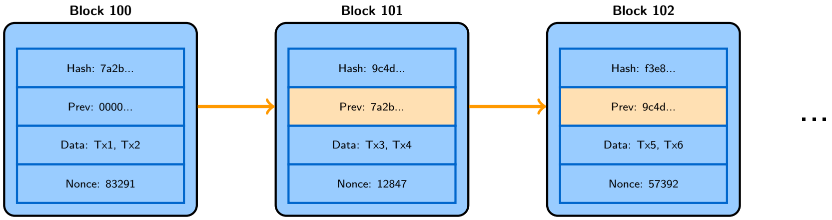
- Can only add data
- Cannot modify history
- Immutable record

Consensus-Based

- Network agrees on state
- No trusted authority
- Rules enforced by code

Simple analogy: A shared Google Doc that everyone can read, only append to, and no one can delete from

Blockchain Structure: Blocks Linked by Hashes



The chain property: Each block contains the hash of the previous block
Why this matters: Change Block 100 → its hash changes → Block 101's "Prev" becomes invalid → cascading invalidity

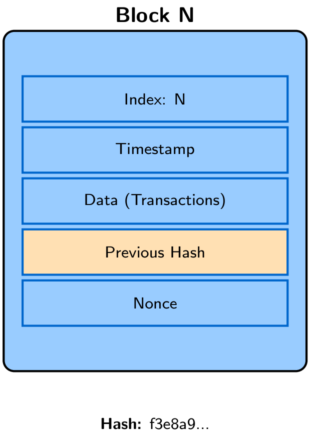
---

This is why blockchain history is considered immutable

Every block contains:

- Index: Position in the chain (Block 0, 1, 2...)
- Timestamp: When the block was created
- Data: Transactions or other content
- Previous Hash: Link to the prior block
- Nonce: Number used for mining
- Hash: The block's unique fingerprint

Key insight: The hash is calculated from ALL other fields.
Change any field, and the hash changes completely.



The Genesis Block: Where It All Begins

What is the Genesis Block?

- The **first block** in any blockchain
- Has no previous block to reference
- "Previous Hash" is typically all zeros
- Created when the blockchain launches

Bitcoin's Genesis Block

- Mined January 3, 2009
- Block 0 – the very first
- Contains a hidden message from Satoshi Nakamoto

Bitcoin Genesis Block Message:

"The Times 03/Jan/2009 Chancellor on brink of second bailout for banks"

This headline proved the block wasn't created earlier and made a political statement about the financial system.

Every blockchain traces back to its genesis block – the root of the chain

Why Tampering Is Detectable

Original Chain	Tampered Chain
Block 100: Hash = 7a2b...	Block 100: Hash = x9f2...
↓ (valid link)	↓ (broken link!)
Block 101: Prev = 7a2b...	Block 101: Prev = 7a2b... X
↓ (valid link)	↓ (broken link!)
Block 102: Prev = 9c4d...	Block 102: Prev = 9c4d... X
✓Valid	X Invalid

To tamper with history, an attacker must:

1. Change the target block's data
2. Recalculate that block's hash
3. Recalculate *every subsequent block's hash*
4. Do this faster than the honest network adds new blocks

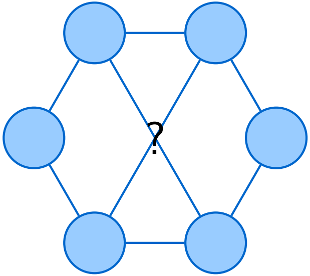
Practically impossible once blocks are deep in the chain

The Challenge
In a decentralized network:

- No central authority
- Nodes don't trust each other
- Messages can be delayed or lost
- Some nodes may be malicious

The Question
How do thousands of strangers agree on:

- Which transactions are valid?
- What order do they go in?
- What is the "true" history?



Who decides the truth?

Consensus Mechanism

A set of rules that allows a distributed network to agree on a single version of truth, even in the presence of faulty or malicious participants.

What Is a Consensus Mechanism?

Definition: A protocol that enables distributed nodes to agree on the state of a shared ledger without requiring trust.

Requirements:

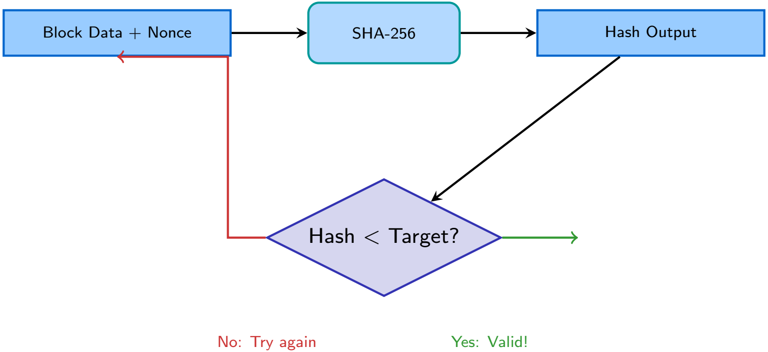
- **Agreement:** All honest nodes accept the same transactions
- **Validity:** Only valid transactions are accepted
- **Termination:** Decisions are eventually made
- **Fault Tolerance:** Works despite some bad actors

Main Approaches:

- **Proof of Work (PoW):** Computational puzzle solving
- **Proof of Stake (PoS):** Economic collateral
- **Proof of Authority:** Trusted validators
- **BFT variants:** Voting-based

Key insight: Different consensus mechanisms make different tradeoffs between security, speed, decentralization, and energy use.

Proof of Work (PoW): Bitcoin's Answer



The "work" in Proof of Work:

- Find a nonce that makes the block hash start with many zeros
- Requires billions of guesses (brute force)
- Easy to verify: just hash once and check
- Hard to produce: requires massive computation

Mining: Finding the Golden Nonce

What miners actually do:

Block + Nonce=1 → Hash: 8f3a2b... X

Block + Nonce=2 → Hash: c91f7e... X

Block + Nonce=3 → Hash: a47d9c... X

...billions more attempts...

Block + Nonce=83291 → Hash: 000000f3... ✓

Key points:

- Nonce = “Number used once” – a value miners keep changing
- Target = How many leading zeros required (sets difficulty)
- Finding a valid hash is like winning a lottery
- Winner gets to add the block and earns rewards

This is why mining requires enormous computational power and electricity

Proof of Work: Security Through Energy

Why It Works

- Attack requires controlling 51% of computing power
- This costs billions in hardware + electricity
- Rational: mining honestly is more profitable
- Economic security: attack cost > potential gain

Advantages

- Battle-tested (Bitcoin since 2009)
- Highly secure
- Truly decentralized

Criticisms

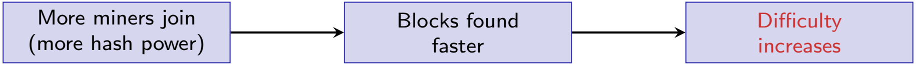
- Enormous energy consumption
- Bitcoin uses more electricity than some countries
- Environmental concerns
- Hardware arms race (specialized ASICs)

Bitcoin Network:
$\approx$ 150 TWh/year
(more than Argentina)

PoW trades energy for security – the "work" is the cost of attack

Difficulty Adjustment: Self-Regulating Security

How does the network maintain consistent block times?



Bitcoin adjusts difficulty every 2,016 blocks (≈2 weeks)
Target: 10 minutes per block on average

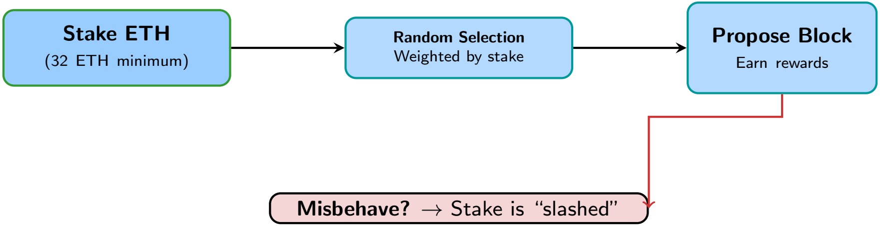
If blocks are too fast:

- Difficulty increases
- Requires more leading zeros
- Harder to find valid hash

If blocks are too slow:

- Difficulty decreases
- Requires fewer leading zeros
- Easier to find valid hash

Proof of Stake (PoS): Ethereum's Answer



The "stake" in Proof of Stake:

- Lock up cryptocurrency as collateral
- More stake = higher chance to be selected as validator
- Honest behavior = earn rewards
- Malicious behavior = lose your stake ("slashing")

How Proof of Stake Works

Step-by-Step Process:

1. **Stake:** Lock cryptocurrency as collateral
2. **Selection:** Protocol randomly selects a validator (weighted by stake – the more coins you lock up, the higher your chance of being selected, like having more lottery tickets)
3. **Propose:** Selected validator creates a new block
4. **Attest:** Other validators verify and vote on the block
5. **Finalize:** Block is added when enough validators agree

Economic Security:

- Attack requires owning 51% of staked coins
- Cheaters lose their stake (slashing)
- "Skin in the game" aligns incentives

Ethereum PoS Stats:

Minimum stake: 32 ETH
Total staked: ∼30M ETH
Active validators: ∼900,000
Energy reduction: ∼99.95%

(vs. Proof of Work)

Slashing Penalties:

- Double signing: Major penalty
- Being offline: Minor penalty
- Coordinated attack: Severe penalty

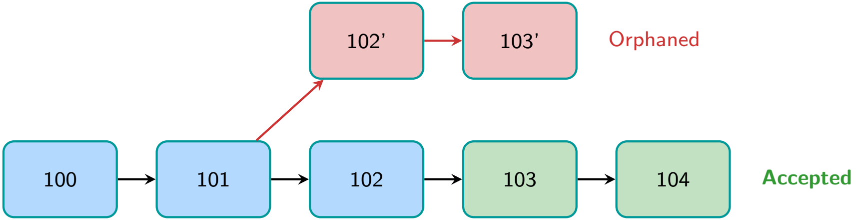
PoW vs. PoS: Head-to-Head Comparison

Aspect	Proof of Work	Proof of Stake
Security basis	Computational power	Economic stake
Energy usage	Very high	Low ($\sim$99.9% less)
Hardware required	Specialized (ASICs)	Standard computers
Entry barrier	High (equipment cost)	High (32 ETH $\approx$ \$60k)
Attack cost	Hardware + electricity	Acquire 51% of stake
Decentralization	Mining pool concentration	Wealth concentration
Finality	Probabilistic	Faster, more definite
Examples	Bitcoin, Dogecoin	Ethereum, Cardano, Solana

Key insight: Neither is “better” – they make different tradeoffs

Ethereum transitioned from PoW to PoS in Sept 2022 (“The Merge”)

How does the network resolve conflicts?



The Rule:

- Nodes accept the chain with **most cumulative work**
- In PoW: typically the longest chain
- In PoS: chain with most validator votes
- Shorter chains become “orphaned”

Why It Works:

- Provides decentralized consensus
- Resolves temporary forks
- Makes 51% attacks expensive
- No voting or coordination needed

Confirmation Depth: How Secure Is Your Transaction?

More confirmations = More security

Your Tx → +1 → +2 → +3 → +4 → +5 → +6

Each additional block makes reversal **exponentially** harder

Confirmations	**Time (BTC)**	**Typical Use**
0 (unconfirmed)	0 min	Very small amounts only
1 confirmation	10 min	Low-value transactions
3 confirmations	30 min	Medium-value transactions
6 confirmations	60 min	**Industry standard for security**

The 51% Attack: Understanding the Risk

What is a 51% attack?
An attacker controlling >50% of network hash power (PoW) or stake (PoS) can:

What they CAN do:

- Double-spend their own coins
- Block other transactions (censorship)
- Reorganize recent blocks
- Prevent confirmations

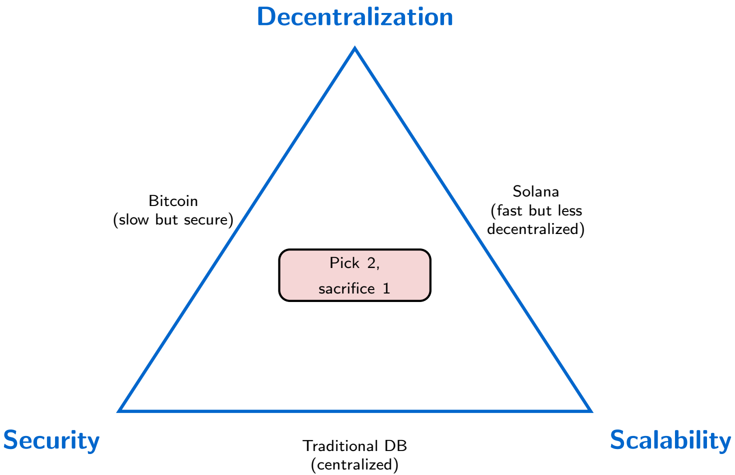
What they CANNOT do:

- Steal other people's coins
- Create coins out of nothing
- Change consensus rules
- Reverse very old transactions

Economic Reality

For Bitcoin, acquiring 51% hash power would cost billions of dollars in hardware and ongoing electricity. The attack would likely destroy the value of the attacker's own coins – making it economically irrational.

The Blockchain Trilemma



The Trilemma Explained

Decentralization

- Many independent validators
- No single point of control
- Censorship resistant
- Geographic distribution

Tradeoff: More nodes = slower

Security

- Resistant to attacks
- Immutable history
- Byzantine fault tolerant
- Economic finality

Tradeoff: More verification = slower

Scalability

- High throughput (Transactions Per Second – TPS)
- Low latency
- Low fees
- Handle demand spikes

Tradeoff: Speed often requires centralization

Performance comparison: Compare how different systems balance speed, certainty, and decentralization

System	TPS	Finality*	Validators
Bitcoin	7	60 min	~15,000 nodes
Ethereum	15-30	12-15 min	~500,000 validators
Solana	65,000	0.4 sec	~1,900 validators
Visa	24,000	seconds	1 (centralized)

*Finality = when a transaction becomes permanent and irreversible

Scaling Solutions: Breaking the Trilemma?

Layer 1 Solutions
Improve the base blockchain

- Larger blocks (more data per block)
- Faster block times
- More efficient consensus
- Sharding (parallel processing)

Examples:

- Ethereum 2.0 (sharding planned)
- Solana (parallel processing)
- Near Protocol (sharding)

Layer 2 Solutions
Build on top of base layer

- Process transactions off-chain
- Settle on main chain periodically
- Inherit security from L1
- Much higher throughput

Examples:

- Lightning Network (Bitcoin)
- Optimism, Arbitrum (Ethereum)
- Polygon (Ethereum)

Key Insight

Layer 2 solutions let blockchains scale WITHOUT sacrificing decentralization or security at the base layer.

Hands-On Exercise: Blockchain Simulation (NB06)

In this notebook, you will:

1. **Build a block** – Create a simple block data structure with all required fields
2. **Create a chain** – Link multiple blocks together using hash pointers
3. **Attempt tampering** – Modify a block and observe chain validation failure
4. **Simulate mining** – Implement a simple Proof of Work algorithm
5. **Adjust difficulty** – See how difficulty affects mining time

Learning Goal: Understand blockchain mechanics through hands-on coding, not just theory.

Open NB06_blockchain_simulation.ipynb in Google Colab

NB06: What You'll Build

Block Class Structure:

- Index, timestamp, data
- Previous hash, nonce
- Calculated hash

Blockchain Class:

- Genesis block creation
- Adding new blocks
- Chain validation
- Tampering detection

Mining Simulation:

- Find valid nonce
- Meet difficulty target
- Measure iterations

Expected Observations:

1. Changing one character breaks the entire chain
2. Mining difficulty dramatically affects time
3. Verification is instant; creation is hard
4. Longer chains resist tampering better

Time estimate: 30-45 minutes
Prerequisites: Basic Python, completed T3.1

Discussion: Blockchain in the Real World

Where does blockchain make sense?

Good Use Cases:

- Permissionless value transfer
- Trustless coordination
- Censorship-resistant applications
- Transparent audit trails
- Programmable money (smart contracts)

Poor Use Cases:

- High-speed trading
- Private data storage
- When trust already exists
- When centralization is acceptable
- When efficiency matters most

Key Question

Before choosing blockchain, ask: "Is the inefficiency worth the trust minimization?"

Discussion: The Energy Debate

Is Proof of Work's energy use justified?

Arguments FOR PoW:

- Energy = security; necessary cost
- Bitcoin increasingly uses renewables
- Traditional finance also uses energy
- Monetary sovereignty has value
- Market decides if cost is worth it

Arguments AGAINST PoW:

- Environmental impact is real
- PoS achieves similar security
- E-waste from mining hardware
- Energy could be used elsewhere
- Alternatives exist and work

The Merge (Sept 2022): Ethereum's switch from PoW to PoS reduced its energy consumption by $\sim$99.95%, proving that high security doesn't require high energy.

Discussion: How Decentralized Is "Decentralized"?

Decentralization is a spectrum, not binary:

Questions to consider:

- How many entities control >50% of validation power?
- Can one entity censor transactions?
- What's the geographic distribution of nodes?
- Who controls the protocol development?

Discussion: The Future of Consensus Mechanisms

What's next for blockchain consensus?

Current Research Areas:

- **Hybrid consensus:** Combine PoW and PoS
- **DAG-based:** IOTA, Hedera
- **BFT variants:** Faster finality
- **Zero-knowledge proofs:** Privacy + scale

Emerging Solutions:

- Rollups (optimistic & ZK)
- Data availability sampling
- Cross-chain bridges

Open Questions:

- Can we truly solve the trilemma?
- How will regulation affect consensus choice?
- Will institutional adoption favor certain mechanisms?
- Can decentralization survive mainstream adoption?

The consensus mechanism debate will shape the future of digital finance.

Executive Summary: Key Takeaways

1. **Blockchain = Distributed + Append-only + Consensus**
   A chain of cryptographically linked blocks maintained by thousands of nodes without central authority.

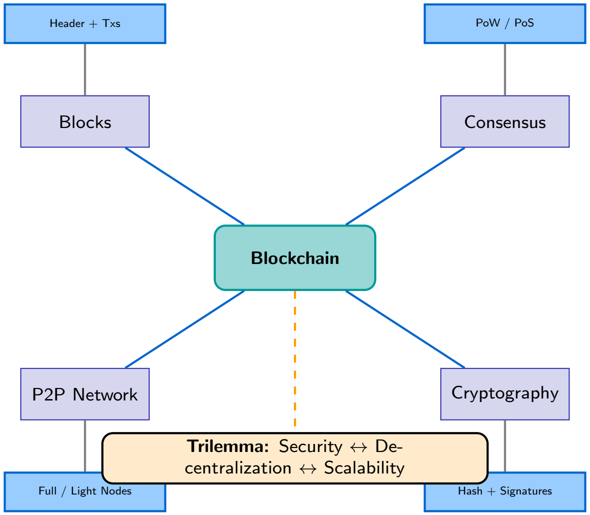
2. **Immutability comes from hash linking**
   Changing any block invalidates all subsequent blocks, making tampering detectable and economically prohibitive.

3. **Consensus mechanisms solve the trust problem**
   PoW uses computational work; PoS uses economic stake. Both create costs for attackers.

4. **The trilemma forces design tradeoffs**
   No blockchain can maximize decentralization, security, AND scalability simultaneously.

5. **Layer 2 solutions offer scalability without compromise**
   Build fast systems on top of secure base layers.

Concept Map: Blockchain Components



Key Terms & Definitions (1/2)

Blockchain A distributed ledger of cryptographically linked blocks maintained by consensus among network participants.

Block A data structure containing transactions, a timestamp, and a reference to the previous block via its hash.

Genesis Block The first block in a blockchain (Block 0), which has no previous block to reference.

Consensus Mechanism A protocol enabling distributed nodes to agree on the state of the ledger without central authority.

Proof of Work (PoW) Consensus mechanism requiring computational effort to find a hash meeting difficulty requirements.

Key Terms & Definitions (2/2)

Proof of Stake (PoS) Consensus mechanism where validators stake cryptocurrency as collateral; misbehavior results in "slashing."

Nonce "Number used once" – a value miners vary to find a valid block hash in Proof of Work.

51% Attack When an attacker controls majority hash power/stake, enabling double-spending or censorship.

Blockchain Trilemma The observation that blockchains can optimize at most two of: decentralization, security, scalability.

Layer 2 Protocols built on top of a base blockchain to increase throughput while inheriting L1 security.

Common Misconceptions

Myth	Reality
"Blockchain data is encrypted"	Blockchain data is **public and transparent**. Anyone can read all transactions. Privacy requires additional layers.
"51% attackers can steal coins"	Attackers can only **double-spend their own coins** or censor transactions. They cannot steal from others.
"PoS is always better than PoW"	Each makes **different tradeoffs**. PoS is more efficient but has different centralization risks.
"Blockchain is infinitely scalable"	The **trilemma is real**. Scaling requires trade-offs or Layer 2 solutions.

Self-Assessment Questions (1/2)

Question 1: Which of the following are components of a block's structure?

A. Only the transaction data and timestamp
B. Index, timestamp, data, previous hash, nonce, and current hash
C. Only the cryptographic hash and signature
D. Public key, private key, and transaction list

Question 1: Which of the following are components of a block's structure?

A. Only the transaction data and timestamp
B. Index, timestamp, data, previous hash, nonce, and current hash
C. Only the cryptographic hash and signature
D. Public key, private key, and transaction list

Answer: B
Explanation: A block contains multiple components: index (position in chain), timestamp (when created), data (transactions/content), previous hash (link to previous block), nonce (proof-of-work number), and hash (the block's digital fingerprint). All these components work together to create a secure, tamper-evident structure.

Question 2: Describe the hash puzzle solving process in PoW mining.

A. Miners solve complex mathematical equations involving prime numbers
B. Miners repeatedly change the nonce and calculate the block hash until finding one that starts with the required number of zeros
C. Miners decrypt encrypted puzzles using private keys
D. Miners compete to find the shortest hash value

Question 2: Describe the hash puzzle solving process in PoW mining.

A. Miners solve complex mathematical equations involving prime numbers
B. Miners repeatedly change the nonce and calculate the block hash until finding one that starts with the required number of zeros
C. Miners decrypt encrypted puzzles using private keys
D. Miners compete to find the shortest hash value

Answer: B
Explanation: The hash puzzle is computationally intensive but simple: find a nonce that produces a hash meeting the difficulty target. Since hash functions are one-way, the only approach is brute force – incrementing the nonce until finding a valid hash.

Question 3: What is the genesis block?

Question 2: Describe the hash puzzle solving process in PoW mining.

A. Miners solve complex mathematical equations involving prime numbers
B. Miners repeatedly change the nonce and calculate the block hash until finding one that starts with the required number of zeros
C. Miners decrypt encrypted puzzles using private keys
D. Miners compete to find the shortest hash value

Answer: B
Explanation: The hash puzzle is computationally intensive but simple: find a nonce that produces a hash meeting the difficulty target. Since hash functions are one-way, the only approach is brute force – incrementing the nonce until finding a valid hash.

Question 3: What is the genesis block?
Answer: The first block in a blockchain (Block 0), with no previous block to reference.

What's Next: Topic 3.3 – Wallets & Transactions

From theory to practice: How users actually interact with blockchains

Topics we'll cover:

- What is a wallet, really?
- Hot vs. cold wallets
- Custodial vs. non-custodial
- Transaction lifecycle
- The UX gap problem

Key insight preview:

Common Misconception:

Wallets don't "store" cryptocurrency.

The blockchain stores balances.

Wallets store *keys* that prove you can spend those balances.

Connection to today: Now that you understand how blockchains work internally, we'll explore how users sign and broadcast transactions to interact with them.

Resources for Further Learning

Essential Reading:

- Nakamoto, S. (2008). *Bitcoin: A Peer-to-Peer Electronic Cash System*
- Buterin, V. (2014). *Ethereum Whitepaper*
- Antonopoulos, A. *Mastering Bitcoin* (Chapters 6-10)

Interactive Tools:

- **Blockchain Demo:** `https://andersbrownworth.com/blockchain/`
- **Bitcoin Block Explorer:** `https://blockstream.info/`
- **Ethereum Block Explorer:** `https://etherscan.io/`

Videos:

- 3Blue1Brown: "But how does bitcoin actually work?"
- MIT OpenCourseWare: Blockchain and Money (Gary Gensler)

NB06 provides hands-on practice with the concepts from this lecture

Questions?

Topic 3.2: Blockchain Mechanics

Consensus, Blocks, and the Trilemma

Next: Topic 3.3 – Wallets & Transactions