Blockchain Foundations
Practice Midterm 1
2026

1. (24 points) True-false questions (no explanations required). 2 points for a correct answer, 0 points for an incorrect answer. 1 point for leaving the answer blank. Knowing you don't know something has value.

   Please shade your answer in completely to receive full credit.

		T	F
(a)	While two conflicting transactions cannot both appear in the same valid block, they can appear in different blocks of a valid chain.	□	■
(b)	A correctly parametrized proof-of-work inequality ensures that all successful queries are always spaced at least Δ apart.	□	■
(c)	As the network delay Δ decreases, the mining target T should be made more difficult.	□	■
(d)	The probability that two different honest miners choose the same nonce to mine with is negligible in κ (see the MINE algorithm in the reference section).	■	□
(e)	The genesis block is anchored at a particular point in time in the real world by including real world data from a newspaper or other publicly verifiable source that cannot easily be faked.	■	□
(f)	Changing the coinbase transaction public key during gossiping will fail because it will invalidate the coinbase signature.	□	■
(g)	While honest miners mine blocks at a bounded rate, an adversary can mine as many blocks as she likes.	□	■
(h)	An adversary who manages to violate ledger safety can issue a transaction spending the money of an honest party.	□	■
(i)	A ledger liveness violation implies a ledger safety violation.	□	■
(j)	An execution with $n = 1$, $t = 0$, $q = 1$ and $\Delta = 1$ sec has no temporary forks whatsoever.	■	□
(k)	In the UTXO model under longest chain rule, a block in the chain can extend multiple parent blocks.	□	■

Reasons:

(a) A block containing a transaction which is conflicting with respect to a predecessor block's UTXO state is invalid.

(b) Successful queries are probabilistic, so they may sometimes be closer than Δ apart.

(c) The target can be made easier as Δ decreases.

(d) Honest miners choose the nonce as a random κ-bit string.

(e) Self-explanatory.

(f) It will fail because the proof-of-work inequality will not be satisfied (coinbase transaction does not have a signature).

(g) The adversary's mining rate is also limited by her hashing power because valid blocks must satisfy the proof-of-work inequality.

(h) The adversary cannot generate signatures because that would require the honest party's secret key.

(i) A censorship attack may violate liveness but not safety.

(j) Since there is only one honest miner, there can't be any forks.

(k) A valid block can have only one parent block.

2. (16 points) Let $H_\kappa : \{0,1\}^* \to \{0,1\}^\kappa$ be a family of collision-resistant hash functions and define $G_{2\kappa} : \{0,1\}^* \to \{0,1\}^{2\kappa}$ as $G_{2\kappa}(x) = H_\kappa(x) \parallel H_\kappa(x)$. Show that $G_{2\kappa}$ is a collision-resistant family of hash functions.

Solution:

We will prove this by contradiction. Suppose that $G_{2\kappa}$ is not a family of collision-resistant hash functions. Then there exists a PPT adversary $\mathcal{A}$ such that

$$\Pr[\textsc{collision}_{G,\mathcal{A}}(2\kappa) = 1] = \mathsf{non\text{-}negl}(\kappa)$$

. We will create a PPT adversary $\mathcal{A}'$ that breaks the collision-resistance of H.

On input 1^κ the adversary $\mathcal{A}'$ works as follows. She invokes $\mathcal{A}$ with input $1^{2\kappa}$ to retrieve x_1, x_2. She then returns x_1, x_2. Since $\mathcal{A}$ is PPT then $\mathcal{A}'$ is PPT. If $\mathcal{A}$ is successful, then $G_{2\kappa}(x_1) = G_{2\kappa}(x_2)$ and $x_1 \neq x_2$. This means that $H_\kappa(x_1) \parallel H_\kappa(x_1) = H_\kappa(x_2) \parallel H_\kappa(x_2)$, and therefore $H_\kappa(x_1) = H_\kappa(x_2)$ and $\mathcal{A}'$ is successful. Therefore,

$$\Pr[\textsc{collision}_{H,\mathcal{A}'}(\kappa) = 1] = \Pr[\textsc{collision}_{G,\mathcal{A}}(2\kappa) = 1] = \mathsf{non\text{-}negl}(\kappa)$$

This contradicts the assumption that H is collision resistant.

Algorithm 1 The reduction.

```
function A'(1^κ)
    x1, x2 ← A(1^{2κ})
    return x1, x2
end function
```

3. (30 points) We are working in a UTXO longest chain system with a block reward of 50 units and a confirmation rule of $k = 6$. Consider the transaction graph illustrated in Figure 1.

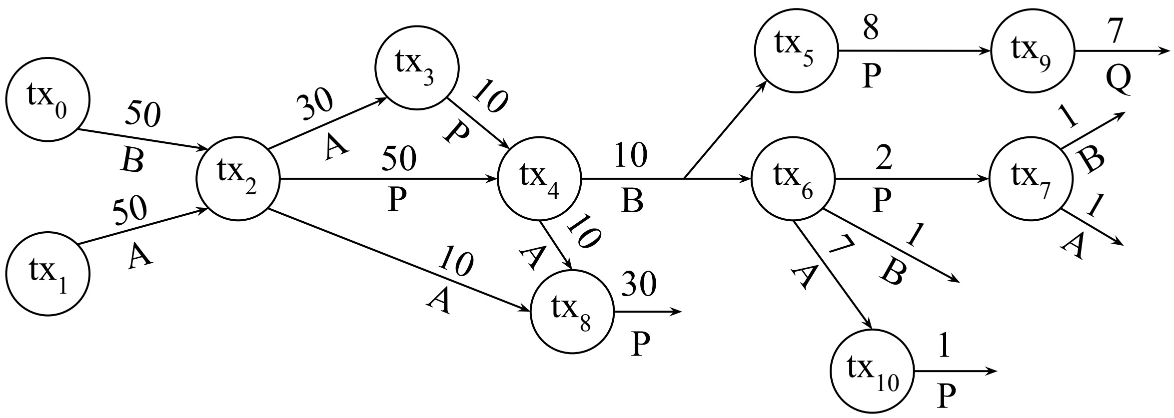

Figure 1: The transaction graph.

The party Q has adopted a chain C_Q such that $\mathbb{L}_Q = (\mathsf{tx}_0, \mathsf{tx}_1, \mathsf{tx}_2)$, while the transactions recorded in C_Q are $(\mathsf{tx}_0, \mathsf{tx}_1, \mathsf{tx}_2, \mathsf{tx}_3, \mathsf{tx}_4)$. The party Q is a miner who collects transactions into a mempool to create a template block to mine on.

(a) (6 points) In what order should the rest of the transactions (tx_5 through tx_{10}) be arranged into a block by Q so that Q's coinbase proceeds are maximized?
Solution: $\mathsf{tx}_6, \mathsf{tx}_7, \mathsf{tx}_{10}$ or $\mathsf{tx}_6, \mathsf{tx}_{10}$.

Possible valid block configurations:

i. $\mathsf{tx}_5, \mathsf{tx}_9$ with the total fee of 3 units

ii. $\mathsf{tx}_6, \mathsf{tx}_7, \mathsf{tx}_{10}$ with the total fee of 6 units

Thus, to maximize coinbase proceeds miner should chose $\mathsf{tx}_6, \mathsf{tx}_7, \mathsf{tx}_{10}$. Transaction tx_7 does not have any fee, so excluding it from the block does not change the coinbase proceeds.

(b) (6 points) What is the output value in Q's new coinbase transaction?

Solution: 56 units: 50 units of the block reward and 6 units of the fees.

(c) (4 points) If Q successfully mined a block in part (a) and the block is buried under $k = 6$ other blocks, what does the new ledger $\mathbb{L}_Q$ report?

Solution: $\mathbb{L}_Q = (\mathsf{tx}_0, \mathsf{tx}_1, \mathsf{tx}_2, \mathsf{tx}_3, \mathsf{tx}_4, \mathsf{tx}_6, \mathsf{tx}_7, \mathsf{tx}_{10})$

(d) (6 points) Which transactions (among tx_5 through tx_{10}) are missing from Q's ledger and why?

Solution: $\mathsf{tx}_5, \mathsf{tx}_8, \mathsf{tx}_9$

i. tx_5 and tx_6 are double-spending transactions, so if tx_6 is in the ledger, tx_5 cannot be included

ii. tx_8 fails the law of conservation, output of 30 units is larger than input 20 units

iii. tx_9 spends the output of tx_5, which is not in the ledger

(e) (4 points) How much unspent money does Q have in the system, if we also include his mining proceeds?

Solution: The transaction tx_9 is not in the ledger, thus, Q only has 56 units of his mining proceeds.

(f) (4 points) How much money do the parties A, B, and P have, provided they are not mining any blocks?

Solution:

i. A has 21 units: 10 units from tx_2, 10 units from tx_4 and 1 unit from tx_7

ii. B has 2 units: 1 unit from tx_6 and 1 unit from tx_7

iii. P has 1 unit from tx_{10}

4. (30 points) Consider the sequence of successful *honest* party queries in Figure 2. Recall that a successful query satisfies the proof-of-work inequality $H(B) < T$.

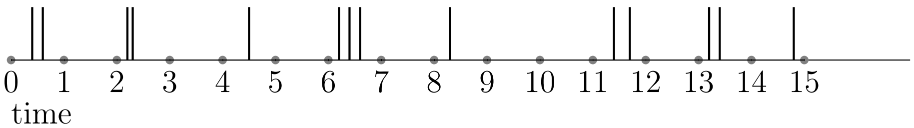

Figure 2: A sequence of honest successful query events.

Consider executions[1] with *maximum* network delay $\Delta = 1$. You are a powerful rushing adversary and you have 7 successful queries at your disposal. You are given the (fictitious) ability to place your successful queries at whichever points on the timeline you prefer. Honest parties gossip blocks, but you can schedule the delay of each honest message freely, as long as it is within a maximum of Δ. As the adversary, you are also allowed to choose how each honest party will choose to break ties among competing chains of the same length.

Describe the following executions, consistent with the above timeline. For each of the below executions, draw the block tree. For each block in the block tree, indicate whether it was honestly or adversarially mined, and what time it was mined at. You can use just the integer part of the time (for example, you can write "1" for a block that was mined at time "1.4"). Your three executions do not all have to be different.

(a) (10 points) An execution in which Common Prefix with $k = 7$ is violated. What is the adopted chain tip of each honest party in your execution?

Solution:

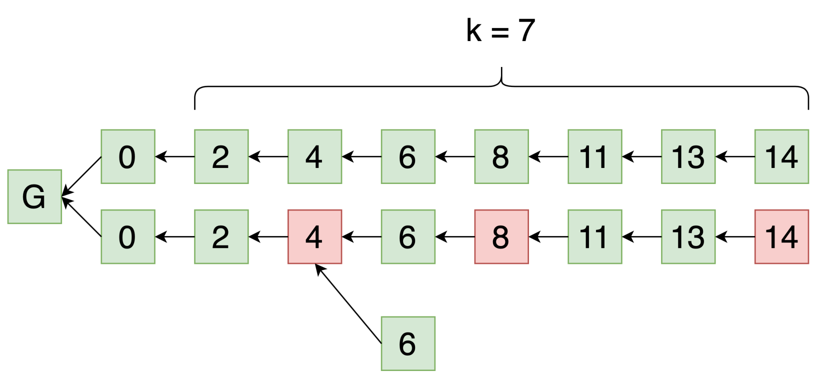

Figure 3: One of executions violating Common Prefix with $k = 7$

[1] An *execution* is the transcript of everything that happened, including *who* mined each block, when each block was mined, what the whole private and public blocktree looks like, when and if each block was broadcast, and when it was received, including all adversarial actions.

(b) (10 points) An execution in which three different honest parties adopt chains C_1, C_2 and C_3 such that $C_1[:-k]$, $C_2[:-k]$, $C_3[:-k]$ are different from each other for $k = 4$. ($C[:-k]$ means the chain resulting from removing the last k blocks in a chain C.)

Solution:

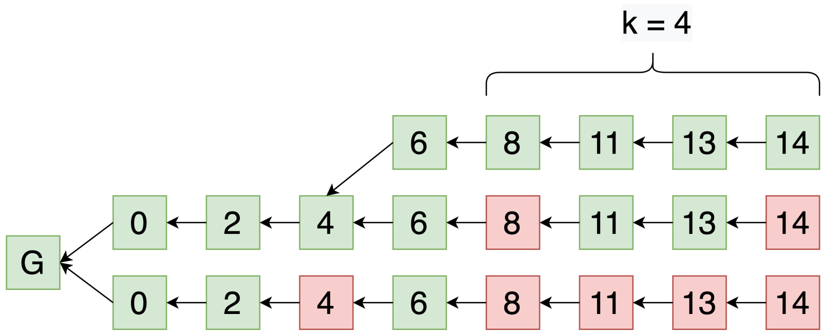

Figure 4: One of the executions violating Common Prefix with $k = 4$

(c) (10 points) An execution *minimizing* Chain Quality (across all executions) of the whole chain for *some* honest party. What is the Chain Quality of the chain adopted by your chosen honest party?

Solution: $CQ = \frac{2}{9}$

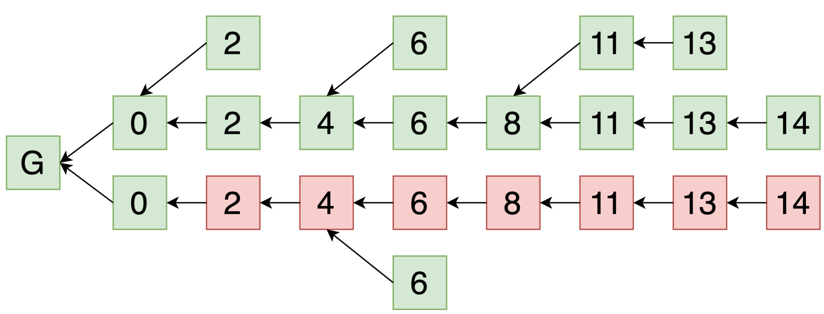

Figure 5: One of the executions minimizing Chain Quality.

Extra page for answers

Reference

Some helpful definitions are provided below.

Definition 1 (Collision Resistance). *A hash function* $H : \{0,1\}^* \longrightarrow \{0,1\}^\kappa$ *is* collision resistant *if for all PPT adversaries* $\mathcal{A}$*,*

$$\Pr[\textsf{\textit{collision-game}}_{H,\mathcal{A}(\kappa)} = 1] = \text{negl}(\kappa)\,.$$

The game is defined in Algorithm 2.

Algorithm 2 The collision-finding game for a hash function H.

function COLLISION-GAME$_{H,\mathcal{A}}(\kappa)$
    $x_1, x_2 \leftarrow \mathcal{A}(1^\kappa)$
    return $H_\kappa(x_1) = H_\kappa(x_2) \wedge x_1 \neq x_2$
end function

Definition 2 (Weak Conservation Law). *A transaction* tx *satisfies the Weak Conservation Law if*

$$\sum_{i \in \mathsf{tx}.\textsf{\textit{ins}}} i.v \geq \sum_{o \in \mathsf{tx}.\textsf{\textit{outs}}} o.v\,.$$

Definition 3 (Velocity). *The* velocity τ *of a chain of an honest party* P *between times* $r_1 < r_2$ *is the ratio* $\frac{|\mathcal{C}^P_{r_2}| - |\mathcal{C}^P_{r_1}|}{r_2 - r_1}$.

Definition 4 (Common Prefix). *A system is said to satisfy* Common Prefix *with parameter* $k \in \mathbb{N}$ *if for all honest parties* P_1, P_2 *and for all times* $r_1 \leq r_2$*, the chains adopted by the honest parties satisfy the property that*

$$\mathcal{C}^{P_1}_{r_1}[:-k] \preceq \mathcal{C}^{P_2}_{r_2}\,.$$

Definition 5 (Chain Quality). *A system is said to satisfy* Chain Quality *with parameters* $\ell \in \mathbb{N}, \mu \in [0,1]$ *if for all honest parties* P *and all times* r*, for all* $i, j \in \mathbb{N}$ *such that* $j - i \geq \ell$*, we have*

$$\frac{|\mathcal{H}(C^P_r[i{:}j])|}{j-i} \geq \mu\,.$$

Definition 6 (Chain Growth). *A system is said to satisfy* Chain Growth *with parameters* $s \in \mathbb{N}, \tau \in \mathbb{R}^+$ *if for all honest parties* P *and all times* $r_1 \leq r_2$ *such that* $r_2 - r_1 \geq s$*, we have*

$$|C^P_{r_2}| - |C^P_{r_1}| \geq \tau(r_2 - r_1)\,.$$

Our variables.

- κ: The security parameter

- $\mathcal{A}$: The uniform PPT adversary
- Π: The honest protocol
- H: The hash function
- G: The genesis block, an *honestly* mined reference block
- Δ: The maximum network delay
- T: The mining target
- p: The probability of a successful query
- n: The total number of parties (includes both honest and adversarial)
- t: The number of adversarial parties
- q: The hashing power of a single party per unit of time
- k: The Common Prefix parameter, in blocks
- μ: The Chain Quality parameter, as a proportion
- τ: The velocity, in blocks per unit of time

Terminology.

- The proof-of-work inequality: $H(B) < T$.
- A *successful query* is a fresh query to the random oracle H that satisfies the proof-of-work inequality.
- A *convergence opportunity* is an *honest* successful query which is spaced at least Δ apart from all other *honest* successful queries.
- A *negligible function* is eventually smaller than all inverse polynomials.
- A block tree has the *Common Prefix* virtue with parameter k if, for any two chains C_1, C_2 currently adopted by honest parties, $C_1[:-k]$ is a prefix of C_2.

Algorithms.
Chain addressing notation.

- $|\mathcal{C}|$: Chain length
- $\mathcal{C}[i]$: i^{th} block in the chain (0-based). The block height is i.
- $\mathcal{C}[-i]$: i^{th} block from the end.
- $\mathcal{C}[0]$: Genesis (by convention honest).

Algorithm 3 The mining algorithm.

```
function Mine(s, x̄)
    ctr ←$ {0,1}^κ
    while true do
        B ← s ‖ x̄ ‖ ctr
        if H(B) < T then
            return B
        end if
        ctr ← ctr + 1
    end while
end function
```

- $\mathcal{C}[-1]$: The tip.
- $\mathcal{C}[i{:}j]$: Chain chunk from block i (inclusive) to j (exclusive).
- $\mathcal{C}[{:}j]$: Chain chunk from the beginning and up to block j (exclusive).
- $\mathcal{C}[i{:}]$: Chain chunk from block i (inclusive) onwards.
- $\mathcal{C}[{:}-k]$: The stable chain.