EE 374 - Blockchain Foundations
Midterm
3:15-4:45pm, May 4, 2022

1. The exam has 4 questions with a total of 100 points. You have 90 minutes to take the exam. Questions have different numbers of points so please allocate your time to each question accordingly.
2. Please write the answer in the designated area underneath each question. If you need more room for your answer, please indicate as such, and continue your response on the blank page at the end of all questions. No additional pages will be allowed.
3. Scratch paper will be provided and collected at the end of the exam, but will not be graded.
4. All answers should be justified, unless otherwise stated.
5. The exam is closed book but you are allowed one double-sided sheet of notes. A list of variables and some other reference information is provided at the end of the exam. No other materials are allowed.

Good luck!

Name:
SUID:

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)	If no one is issuing any transactions, the mempool will be empty.	■	☐
(c)	A correctly parametrized proof-of-work equation ensures that all successful queries are always spaced at least Δ apart.	☐	■
(d)	As the network delay Δ decreases, the mining target T should be made more difficult.	☐	■
(e)	The probability that two different honest miners choose the same nonce to mine with is negligible in κ.	■	☐
(f)	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.	■	☐
(g)	Changing the coinbase transaction public key during gossiping will fail because it will invalidate the coinbase signature.	☐	■
(h)	While honest miners mine blocks at a bounded rate, an adversary can mine as many blocks as she likes.	☐	■
(i)	An adversary who manages to violate ledger safety can issue a transaction spending the money of an honest party.	☐	■
(j)	A ledger liveness violation implies a ledger safety violation.	☐	■
(k)	An execution with $n = 1$, $t = 0$, $q = 1$ and $\Delta = 1$ sec has no temporary forks whatsoever.	■	☐
(l)	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 its previous block's UTXO state is invalid.

(b) Mempool is formed by transactions broadcast to the network.

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

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

(e) Since honest miners choose nonce as a random κ-bit string.

(f) Self-explanatory.

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

(h) Adversary's mining rate is also limited by her hashing power because valid blocks must satisfy the proof-of-work equation.

(i) Adversary cannot generate signatures because that would require the honest party's secret key.

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

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

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

2. (16 points) Let $H : \{0,1\}^* \to \{0,1\}^\kappa$ be a collision-resistant hash function and define $G(x) = H(H(x))$. Show that G is a collision-resistant hash function.

Solution:

We will prove this by contradiction. Suppose that G is not collision-resistant. Then there exists a PPT adversary $\mathcal{A}$ such that $\Pr[\text{COLLISION}_{G,\mathcal{A}}(\kappa) = 1] = \mathsf{non\text{-}negl}(\kappa)$. We will create a PPT adversary $\mathcal{A}'$ that breaks the collision-resistance of H, that is, $\Pr[\text{COLLISION}_{H,\mathcal{A}'}(\kappa) = 1] = \mathsf{non\text{-}negl}(\kappa)$.

On input 1^κ the adversary $\mathcal{A}'$ works as follows. She invokes $\mathcal{A}$ with input 1^κ to retrieve x_1, x_2. She checks if $H(x_1) = H(x_2)$. If so, she returns x_1, x_2. If not, she returns $H(x_1), H(x_2)$.

If x_1, x_2 is a collision for G, i.e. $x_1 \neq x_2$ and $H(H(x_1)) = H(H(x_2))$, there are two cases:

- If $H(x_1) = H(x_2)$, then x_1, x_2 is a collision for H.
- If $H(x_1) \neq H(x_2)$, then $H(x_1), H(x_2)$ is a collision for H.

Therefore, in the event that $\mathcal{A}$ has found a collision, $\mathcal{A}'$ has also found a collision. Furthermore, $\mathcal{A}'$ runs in polynomial time. We have that $\Pr[\text{COLLISION}_{H,\mathcal{A}'}(\kappa) = 1] = \Pr[\text{COLLISION}_{G,\mathcal{A}}(\kappa) = 1]$.

Since $\Pr[\text{COLLISION}_{G,\mathcal{A}}(\kappa) = 1]$ is non-negligible in κ, then $\Pr[\text{COLLISION}_{H,\mathcal{A}'}(\kappa) = 1]$ is also non-negligible in κ. This contradicts the assumption that H is collision resistant.

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.

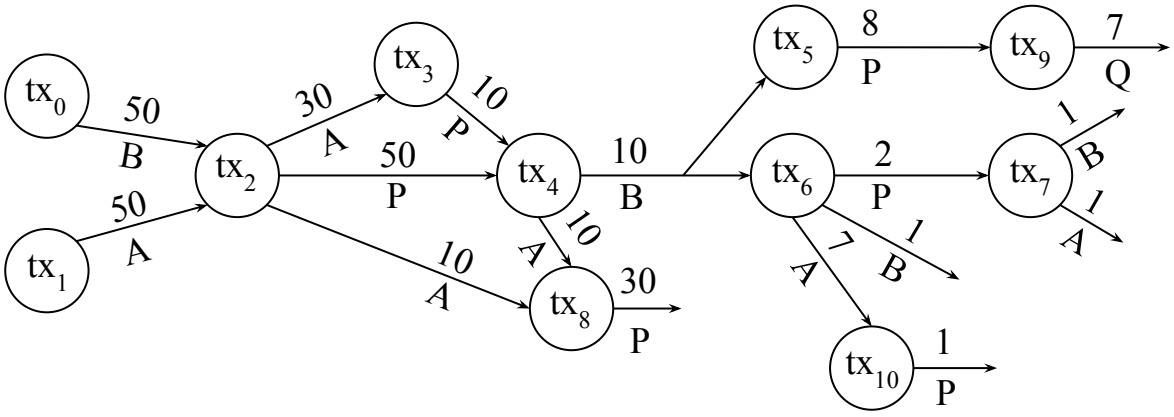

Figure 1: The transaction graph.

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 equation $H(B) \leq T$.

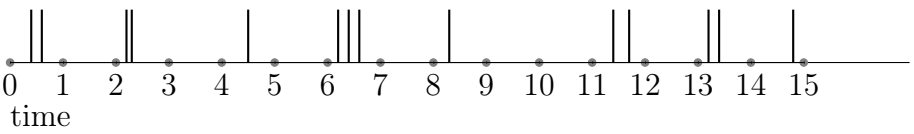

Figure 2: A sequence of honest successful query events.

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:

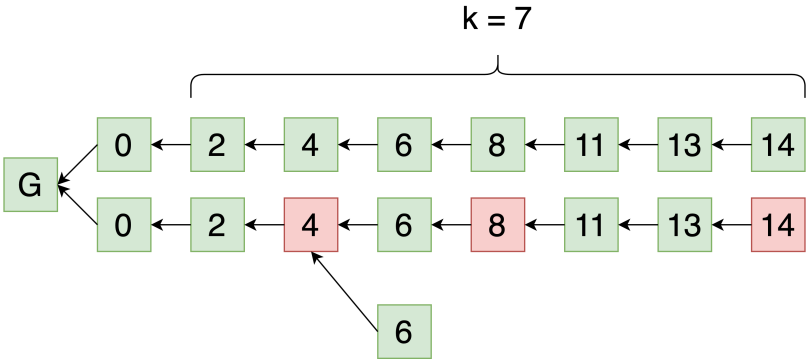

Figure 3: One of executions violating Common Prefix with $k = 7$

[1]Recall that 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:

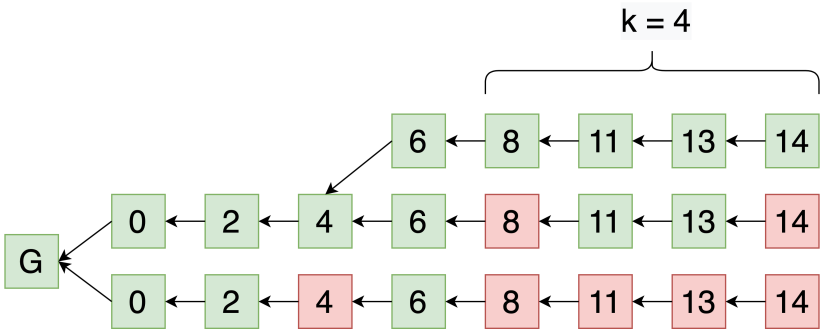

Figure 4: One of the executions violating Common Prefix with $k = 4$

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

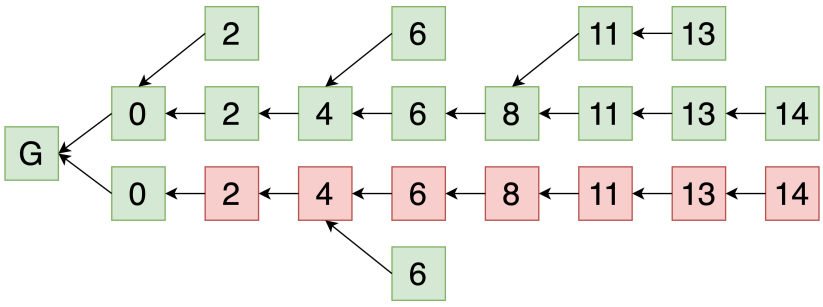

Figure 5: One of the executions minimizing Chain Quality.

(d) (Bonus: 6 points) Prove that no other execution can yield a smaller Chain Quality than the one in part (c).

Solution: Including the Genesis block, the honest parties mined blocks in 9 slots in total. The adversary mined blocks in 7 slots at most. Thus, the Chain Quality cannot be lower than $1 - \frac{7}{9} = \frac{2}{9}$.

Extra page for answers

Reference

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 Chain Growth parameter, in blocks per unit of time

Terminology.

- The proof-of-work equation: $H(B) \leq T$.
- A *successful query* is a fresh query to the random oracle H that satisfies the proof-of-work equation.
- 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.

Algorithms.

Algorithm 1 The mining algorithm.

function $\textsc{Mine}(s, \overline{x})$
    $ctr \xleftarrow{\$} \{0,1\}^\kappa$
    while true **do**
        $B \leftarrow s \,\|\, \overline{x} \,\|\, ctr$
        if $H(B) \leq T$ **then**
            return B
        end if
        $ctr \leftarrow ctr + 1$
    end while
end function

Algorithm 2 The collision resistance game.

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