Checkup 2

Your Name: _____ **UVa Computing ID:** _____

For this quiz, you should **work alone**. You may not use any resources other than your own brain, body, and a simple writing instrument.

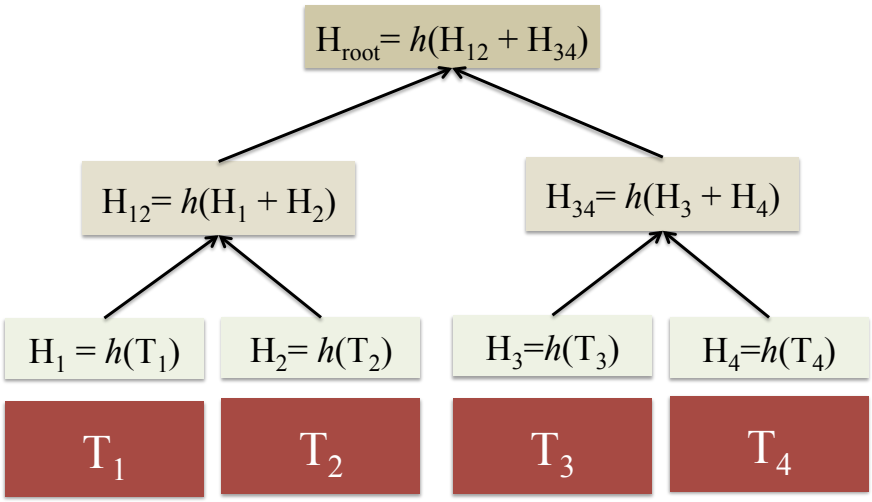
For each question, write a correct, clear, insightful, and delightful answer in the space provided. The space provided is more than sufficient for a full-credit answer, but if you need to use additional space, make sure your answers are clearly marked.

1. (from Notes 3) How are digital signatures and real-life (physical) signatures similar and different?

2. (from Notes 6) What properties are needed for a function to be a cryptographically secure hash function?

3. The security of bitcoin depends on no one discovering an easy way to solve any one of several believed-to-be-hard problems. Identify and clearly state *two* problems that must be hard for bitcoin to be secure, and explain what a malicious person with an efficient way of solving each problem could do to wreak havoc on the bitcoin economy.

4. For the Merkle tree shown below, what is the minimum amount of data needed to verify T_3? (just list the values needed)



Recall the bitcoin block header:

Label	Bytes	Description
`version`	4	Block version information
`prev_block`	32	Hash of the previous block
`merkle_root`	32	Hash of Merkle tree of all transactions
`timestamp`	4	When block was created (overflows in 2106)
`bits`	4	Difficulty target used for this block
`nonce`	4	Nonce found to generate this block

5. Supposed instead of including the hash of the previous block (`prev_block`) in the block header, bitcoin just used the block number of the previous block. Would this be okay? (Either explain why it provides similar security properties to what Bitcoin does, or explain what problems it would cause.)

6. Supposed instead of including the hash of the previous block in the block header, bitcoin just used the Merkle hash of previous block. Would this be okay? (Either explain why it provides similar security properties to what Bitcoin does, or explain what problems it would cause.)

7. (Bonus) Would it be possible to design a cryptocurrency blockchain where the block is just the 32-byte Merkle root? That is, the `version`, `prev_block`, `timestamp`, `bits`, and `nonce` fields are all removed, but additional constraints may be placed on the Merkle tree used to record a block. Either explain how to design a secure ledger using just the Merkle roots as the block headers, or argue that more fields are necessary to provide a secure and effective blockchain.