EE374 Blockchain Infrastructure **Stanford, Spring 2025**

Homework #2 Solutions

Due: Fri, April-18-2025, 11:59pm – Gradescope entry code: `R57ZN7`

Please upload your answers timely to Gradescope. Start a new page for every problem. We strongly suggest LaTeX to type your answers. For the programming/simulation questions you can use any reasonable programming language (please no assembly, brainfuck, etc. ☺). Comment your source code and include the code and a brief overall explanation with your answers. A tentative point distribution (in % of the total) is provided in brackets. For most problems there is more than one valid way of solving them!

1. (20%) Hash functions.
   a) State the random oracle model for hash functions.
   b) Look at the following hash of a recent Bitcoin block:
   000000000000000000009942393c797d6d0bff9e88bc1ae84a3d7e7546f46f61
   How many attempts did the network approximately require to mine this block under the random oracle model?
   c) State the collision resistance property of hash functions.
   d) Argue that a hash function under the random oracle model is collision resistant.

Answer:

a) When a hash function modeled as a random oracle is queried with a new input, a random value from its output distribution is sampled and returned. Upon receiving the same input, the hash function returns the same output value.

b) It requires approximately 2^{80} attempts.

c) Collision resistance states that given a keyed hash function $H(k, \cdot)$ and the key k, it is computationally difficult to find two distinct messages m_0 and m_1 such that $H(k, m_0) = H(k, m_1)$.

d) If a hash function with a large output space, e.g., $\{0,1\}^{256}$, behaves like a random oracle, then the probability that two distinct inputs produce the same output is extremely low, i.e., $1/2^{256}$.

---

2. (10%) Explain the key differences between the accounts model and Bitcoin's UTXO model.

Answer:

The accounts model maintains a state, where each user has a unique account storing their current balance. Transactions modify the balance directly by updating the account associated with the sender(s) and recepient(s). Therefore, blockchains with the accounts model must take extra precaution to prevent 'replay' attacks.

In contrast, the UTXO (unspent transaction output) model uses a single entry per coin rather than per user as in the accounts model. Users can own multiple coins and must refer to the specific coins when spending them. Each transaction consumes the 'spent' coins, which cannot be used in future transactions, and creates new coins.

---

3. (20%) Answer the following questions.

   a) **What's the output of the following script if the input is 7? What function does it implement?**

```
<7>
OP_DUP
OP_DUP
OP_ADD
OP_DUP
OP_ADD
OP_ADD
```

   b) **What's the output of the following script? What function does it implement?**

Answer:

The answer is 35. It implements $f(x) = 5x$.

```
<11>
<3>
OP_2DUP
OP_MAX
```

```
OP_TOALTSTACK
OP_MIN
OP_FROMALTSTACK
```

Answer:

The answer is '11 3', where 11 is at the top of the stack. It takes two numbers and orders them from largest to smallest on the stack: $f([x, y]) = [\max(x, y), \min(x, y)]$, where for a given vector $[x_1, \ldots]$, x_1 denotes the top stack item.

Hint: You can use https://ide.scriptwiz.app to verify your answers.

4. (15%) Parse the following Bitcoin transaction and write it in the format stated here https://gist.github.com/RobinLinus/7971daeb88ecd939825bc456721d9378

```
01000000018e70ee5afa829e776d006cdb340c79029e95d9bdd059751590196
97172e944ea000000008a4730440220694ff325724a4f4b0f3f0c36bf8e94ca
c58ad7c9b4d5bd8c7286c0da623f0b2c02206ae94680a8f31f30cd846da258e
919c94afe2dd629b4f4ce11bbe8165ff99a5f014104fc60372d27b067ca306b
a812ced9c8cd69296b83a40b9b57c593258c1b9e0ee1c0c621ca558b878395f
9645a4b67a96e51843e9c060d43a3833fdd29a91f4f31ffffffff0200e1f505
000000001976a914f369e8330a1e9a349721c3d790ae4d38b68e525288ac00e
9a435000000001976a914f10eb3bfce5ab24537e571ceb3862d2949f7c5e288
ac00000000
```

Hint: You can find the data model of Bitcoin transactions here https://gist.github.com/RobinLinus/7971daeb88ecd939825bc456721d9378

Answer:

```
version: 01000000
inputsCount: 01
input #1:
    txid: 8e70ee5afa829e776d006cdb340c79029e95d9bdd05975159019697172e944ea
 vout: 00000000
 scriptSigSize: 8a
 scriptSig:
    OP_PUSHBYTES_71: 47
    data: 30440220694ff325724a4f4b0f3f0c36bf8e94cac58ad7c9b4
        d5bd8c7286c0da623f0b2c02206ae94680a8f31f30cd846da258e919
```

```
        c94afe2dd629b4f4ce11bbe8165ff99a5f01
       OP_PUSHBYTES_65: 41
       data: 04fc60372d27b067ca306ba812ced9c8cd69296b83a40b9b57c
        593258c1b9e0ee1c0c621ca558b878395f9645a4b67a96e51843e9c06
        0d43a3833fdd29a91f4f31
 sequence: ffffffff
outputsCount: 02
output #1:
 value: 00e1f50500000000
 scriptPubKeySize: 19
    scriptPubKey:
        OP_DUP: 76
        OP_HASH160: a9
        data: 14f369e8330a1e9a349721c3d790ae4d38b68e5252
        OP_EQUALVERIFY: 88
        OP_CHECKSIG: ac
output #2:
 value: 00e9a43500000000
 scriptPubKeySize: 19
    scriptPubKey:
        OP_DUP: 76
        OP_HASH160: a9
        data: 14f10eb3bfce5ab24537e571ceb3862d2949f7c5e2
        OP_EQUALVERIFY: 88
        OP_CHECKSIG: ac
locktime: 00000000
```

---

5. (35%) Consider a mining pool *F2Pool* that has 20% of the total hash rate of the current Bitcoin network.

   a) What is the distribution of the time we need to wait until a block is mined by *F2Pool*? Justify your answer.

Answer:

The rate at which *F2Pool* computes hashes is 20% of the hash rate of the total network. *F2Pool*'s mining successes are independent of the other mining that goes on in the network. Thus the time taken for *F2Pool* to mine their next block has an exponential distribution with rate $\lambda = 0.2 \cdot \frac{1}{600} = \frac{1}{3000}$. The probability density function is $\frac{1}{3000} e^{-t/3000}$ and the mean is 3000 s.

---

b) What is the probability that the next block mined is from *F2Pool*? Justify your answer.

Answer:

From 3.a), the time taken for *F2Pool* to mine their next block has an exponential distribution with rate $\frac{1}{3000}$. Since the rest of the network has 80% of the hash rate, the time taken for the rest of the network (except *F2Pool*) to mine their next block has an exponential distribution with rate $0.8 \cdot \frac{1}{600} = \frac{1}{750}$. Define two random variables $T_{F2} \sim \text{Exponential}\left(\frac{1}{3000}\right)$, and $T_{\text{other}} \sim \text{Exponential}\left(\frac{1}{750}\right)$. Then the probability that the next block is mined by *F2Pool* is

$$\begin{aligned}
\Pr\left(T_{F2} < T_{\text{other}}\right) &= \int_0^\infty \Pr\left(T_{F2} < t\right) f_{T_{\text{other}}}(t)\, dt \\
&= \int_0^\infty \left(1 - e^{-t/3000}\right) \frac{1}{750} e^{-t/750}\, dt \\
&= 1 - \frac{1}{750} \cdot \frac{1}{\frac{1}{750} + \frac{1}{3000}} = 0.2
\end{aligned}$$

Alternatively, every miner tries nonces randomly to mine a block, so each attempted nonce is independently and equally likely to mine the next block. Since *F2Pool* performs 20% of the hashes, the probability that the next block is mined by *F2Pool* is 20%.

---

c) At the end of Lecture 2, we consider an attack on the confirmation rule where a transaction enters the ledger as soon as its block enters the longest chain. Alice's transaction is in a block B and she is trying to mine two blocks in private before the honest miners are able to mine a new block on B, so as to remove the transaction from the ledger. Suppose *F2Pool* is trying to perform this attack. What is the probability it will succeed? You can assume the attacker starts mining the private blocks as soon as the transaction enters the mempool, and that the transaction will be in the next block mined by the network. Make explicit any other assumptions you are making in deriving the answer.

Answer:

Consider that the block B was mined by an honest miner (in this case, someone other than *F2Pool*). Out of the 3 blocks mined right after the transaction enters the mempool, one of them (B) is mined by the honest miners. If the attacker mines the other 2 blocks, the attack will be successful. This is because the attacker can build

a private chain using these 2 blocks which will then replace B. The attack is shown in Fig. 1. The attack will be successful no matter in which order these 3 blocks are mined, as long as no other honest block is mined before the 2 adversarial blocks. Since there is a 20% probability that the next block will be mined by *F2Pool*, the probability of this event is $3 \cdot (0.2)^2 \cdot 0.8 = 0.096$.

More generally, the adversary can replace their transaction even if B was mined by the adversary. They first add the transaction to B and make B public to confirm the transaction, then make the private chain public, to replace the transaction. In this case, the adversary mines all 3 blocks after the transaction enters the mempool. Therefore, the probability that the attack is successful is $3 \cdot (0.2)^2 \cdot 0.8 + (0.2)^3 = 0.104$.

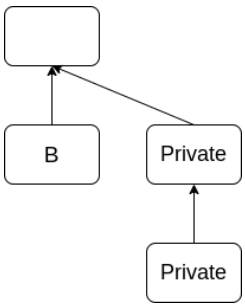

Figure 1: Private attack in 3c. The private blocks will be released as soon as the transactions in B have been added to the ledger.