EE374 Blockchain Infrastructure **Stanford, Spring 2025**

Homework #1

Due: Fri, April-11-2025, 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%)Nakamoto mentioned the word ”trust” multiple times in his P2P Foundation post.

   a) What's wrong with ”trust”?

   b) Explain in what sense is a secure digital signature scheme ”trustless”?

   c) Is Nakamoto's proof-of-work longest chain protocol trustless? If not state the trust assumptions for the protocol. How do they differ from the trust assumptions of a traditional currency?

2. (30%) In this question we discuss the stochastic modelling of the mining times of Bitcoin.

   a) A reasonable model for the distribution of the time between two consecutive mining events is exponential. Write down this probability density function with specific choice of all parameters to model ideal Bitcoin.

   b) What is the standard deviation of the inter-mining time under this model? What is the ratio of the standard deviation over the mean?

   c) What is the mean of the time it takes to mine 10 blocks? What is the standard deviation, and the ratio of the standard deviation over the mean? Be explicit about any assumptions you made to get this conclusion, and justify your modeling assumptions.

   d) Using data from `https://btc.com/block`, estimate the standard deviation of the inter-block mining time. Is it close to what your model in part (a) predicts? Explain if there is any significant discrepancy. State carefully how you perform the estimation and justify why you estimate this way.

3. (20% The total hashrate of the Bitcoin network on January 1, 2018 was 14.4 EH/s.

   a) Estimate the threshold in the hash inequality on that day from this fact. Compare this with the true threshold. Why might there be a discrepancy?

b) Assume all mining was conducted using back then state-of-the-art Antminer S9 hardware, which delivers 14 TH/s at a power consumption of 1372 W. What was the power consumption of the Bitcoin network? Find a comparable country in `https://en.wikipedia.org/wiki/List_of_countries_by_electricity_consumption`.

4. (30%) Let $H : \{0,1\}^k \to \{0,1\}^k$ denote a cryptographic hash function, and define the following signature scheme with security parameter k:

   - $\mathsf{G}()$: Choose 2ℓ elements (each consisting of k bits) randomly from the set $\{0,1\}^k$: $(x_{1,0}, x_{1,1}, \ldots, x_{\ell,0}, x_{\ell,1}) \leftarrow (\{0,1\}^k)^{2\ell}$. Find $y_{i,j} = H(x_{i,j})$ for all $i \in \{1, \ldots, \ell\}$ and $j \in \{0,1\}$. Then, output

   $$(\mathsf{pk} \triangleq (y_{1,0}, y_{1,1}, \ldots, y_{\ell,0}, y_{\ell,1}), \mathsf{sk} \triangleq (x_{1,0}, x_{1,1}, \ldots, x_{\ell,0}, x_{\ell,1}))$$

   - $\mathsf{S}(\mathsf{sk}, m \in \{0,1\}^\ell)$: Parse sk as $(x_{1,0}, x_{1,1}, \ldots, x_{\ell,0}, x_{\ell,1})$. To sign a message $m \in \{0,1\}^\ell$, first parse m as a bit string $m = (b_1, \ldots, b_\ell)$, where each b_i is a bit. Then, output the signature $\sigma \triangleq (x_{1,b_1}, \ldots x_{\ell,b_\ell})$.
   - $\mathsf{V}(\mathsf{pk}, m, \sigma)$: Parse pk as $(y_{1,0}, y_{1,1}, \ldots, y_{\ell,0}, y_{\ell,1})$, σ as $(\hat{x}_1, \ldots, \hat{x}_\ell)$ and the message m as a bit string $m = (b_1, \ldots, b_\ell)$. If $H(\hat{x}_i) = y_{i,b_i}$ for all $i \in \{1, \ldots, \ell\}$, then output 1. Else, output 0.

   To make the signature scheme concrete, let's walk through a small example. Suppose $k = 256$ and $\ell = 4$. In this case, $\mathsf{G}()$ randomly samples some values $x_{1,0}, x_{1,1}, x_{2,0}, x_{2,1}, x_{3,0}, x_{3,1}$, $x_{4,0}$, and $x_{4,1}$ from $\{0,1\}^{256}$, and outputs

   $$\mathsf{sk} \triangleq (x_{1,0}, x_{1,1}, x_{2,0}, x_{2,1}, x_{3,0}, x_{3,1}, x_{4,0}, x_{4,1})$$

   and

   $$\mathsf{pk} \triangleq (y_{1,0}, y_{1,1}, y_{2,0}, y_{2,1}, y_{3,0}, y_{3,1}, y_{4,0}, y_{4,1}),$$

   where

   $$y_{1,0} \triangleq H(x_{1,0}),\ y_{1,1} \triangleq H(x_{1,1}), y_{2,0} \triangleq H(x_{2,0}),\ y_{2,1} \triangleq H(x_{2,1})$$
   $$y_{3,0} \triangleq H(x_{3,0}),\ y_{3,1} \triangleq H(x_{3,1}),\ y_{4,0} \triangleq H(x_{4,0}),\ y_{4,1} \triangleq H(x_{4,1})$$

   Now, as the signature σ_m for the message $m = (0,1,0,0)$, the signing algorithm S outputs $\sigma_{m=(0,1,0,0)} = (x_{1,0}, x_{2,1}, x_{3,0}, x_{4,0})$. Upon receiving the public key pk, the message m and the signature σ_m, the verification algorithm V checks if $y_{1,0} = H(x_{1,0})$, $y_{2,1} = H(x_{2,1})$, $y_{3,0} = H(x_{3,0})$ and $y_{4,0} = H(x_{4,0})$.

   Please answer the following questions:

   a) In the small example above, what is the signature for the message $m = (1,1,1,0)$? Does this signature verify given the message m (i.e., does V output 1 upon receiving pk, m and this signature)?

   b) For a given k and ℓ (some polynomial of k), is the signature scheme correct? Explain.

c) For a given k and ℓ (some polynomial of k), is this signature scheme secure? If so, argue briefly. If not, describe an adversary $\mathcal{A}$ that wins the chosen message attack game against the challenger (recall the security definition for signature schemes).

d) If each pair of keys $(\mathsf{pk}, \mathsf{sk})$ were to be used to sign a *single* message, would this signature scheme be secure? If so, argue briefly. If not, describe an adversary $\mathcal{A}$ that wins the chosen message attack game against the challenger (recall the security definition for signature schemes).

This is called a Lamport signature, named after Leslie Lamport (also an author of the paper 'The Byzantine Generals Problem').