User manual and instructions for Coin Flipping by Mail

Abstract

This document serves as a guide for the "Coin Flipping by Mail" application. It includes a brief theoretical introduction and provides a step-by-step tutorial for using the application.

1 Initial Situation

Two individuals want to perform a coin toss. However, they are not in the same location and cannot verify the statements of the person tossing the coin. The goal of this application is to make the coin toss tamper-proof and transparent for both individuals, enabling the coin toss to be conducted in different locations.

2 Theoretical Foundations

To ensure a fair coin flip, it is essential to consider several theoretical aspects. The following foundational theories will be briefly discussed:

- Commitment Schemes
- Hash Functions

2.1 Commitment Schemes

A Commitment Scheme is a cryptographic protocol that allows committing to information without immediately revealing it. It consists of two phases: Commit and Reveal. In the Commit phase, the sender commits to information by creating an encrypted version of it. In the Reveal phase, the sender can subsequently disclose the original information, which must match the commitment made earlier.

A commitment scheme must satisfy two fundamental properties. First, the chosen value must not be derivable from the commitment message or its probability detectable. This property is called hiding. Second, the chosen value must not be alterable without detection once it is revealed in the opening phase. This property is called binding. [Cap23]

2.2 Hash Functions

Hash functions are mathematical algorithms that transform an input (or message) of arbitrary length into a fixed-size output value called a hash, or hash value.
This hash value is typically a string of a fixed length and represents the input data. A crucial feature of a good hash function is that small changes in the input result in drastic changes in the hash value (avalanche effect).
Hash functions are deterministic, meaning the same input will always produce the same hash value. They are a fundamental component in various areas of computer science, such as data integrity, digital signatures, and cryptocurrencies. Another important property is one-wayness, where it should be practically impossible to reconstruct the original input from the hash value. Additionally, good hash functions should be collision-resistant, meaning it should be very difficult to find two different inputs that produce the same hash value.[CLRS09]

3 Step-by-Step tutorial

Preliminary Note: This guide describes a coin toss game played from the perspective of Alice and Bob. Alice performs the coin toss, and Bob tries to guess the result – heads or tails.

The protocol consists of three phases:

- 3.1 Phase 1 – Generate the hash value and send it to Bob
- 3.2 Phase 2 – Bob guesses the value of the coin
- 3.3 Phase 3 – Reveal phase and Bob's verification option

3.1 Phase 1 – Generate the hash value and send it to Bob

After the application has started, the control panel appears (see Figure 1).

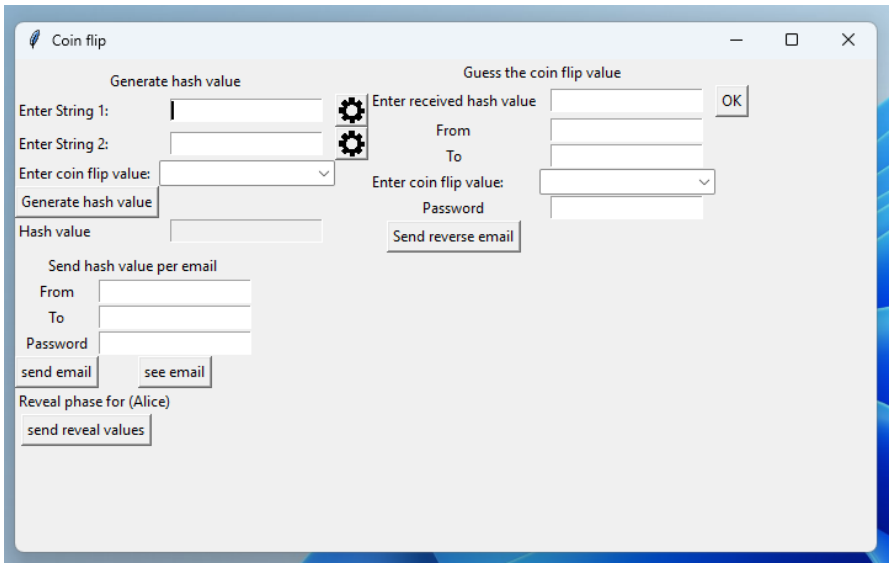

Figure 1: control panel

Now Alice can flip a coin and select the result of the toss from the drop-down menu (see Figure 2).

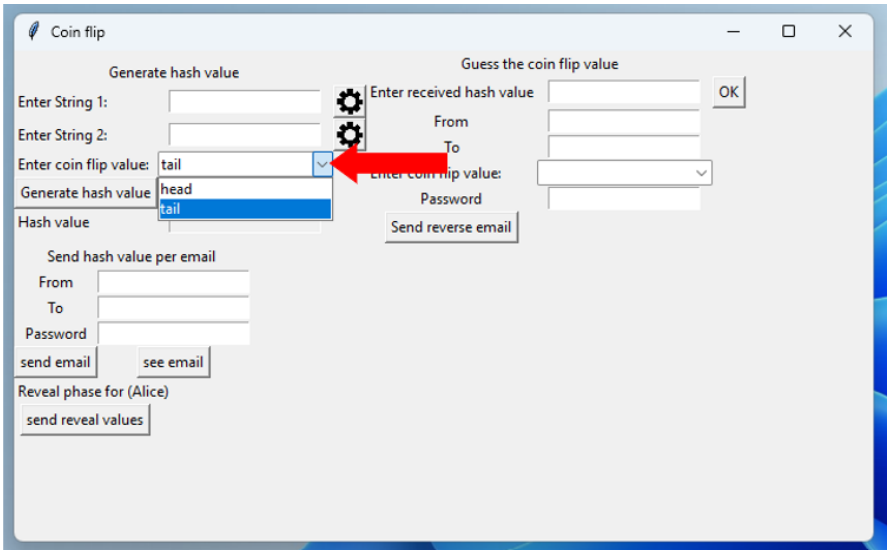

Figure 2: select the result of the toss

Next, two random strings are generated by clicking on the gear icons. (see Figure 3)

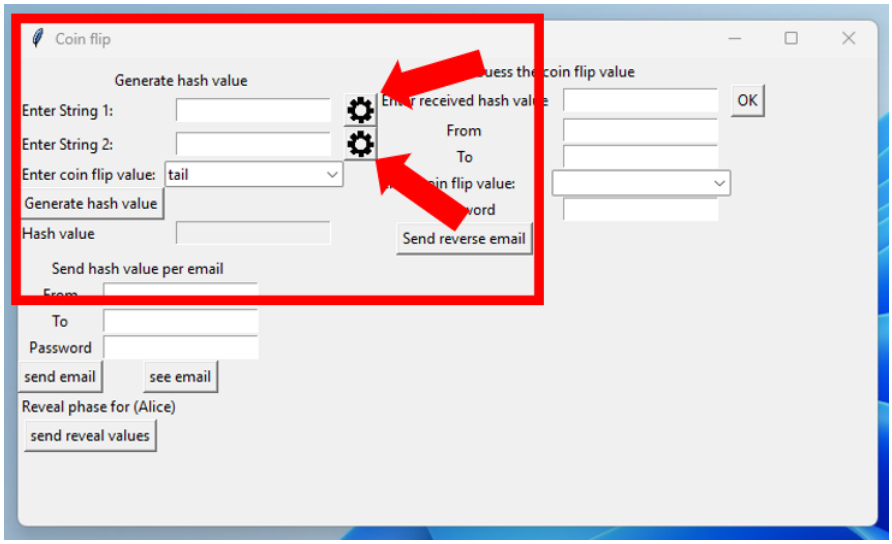

Figure 3: Generate random strings

From the previous inputs, Alice now generates a hash value by clicking the "Generate hash value" button. (see Figure 4)

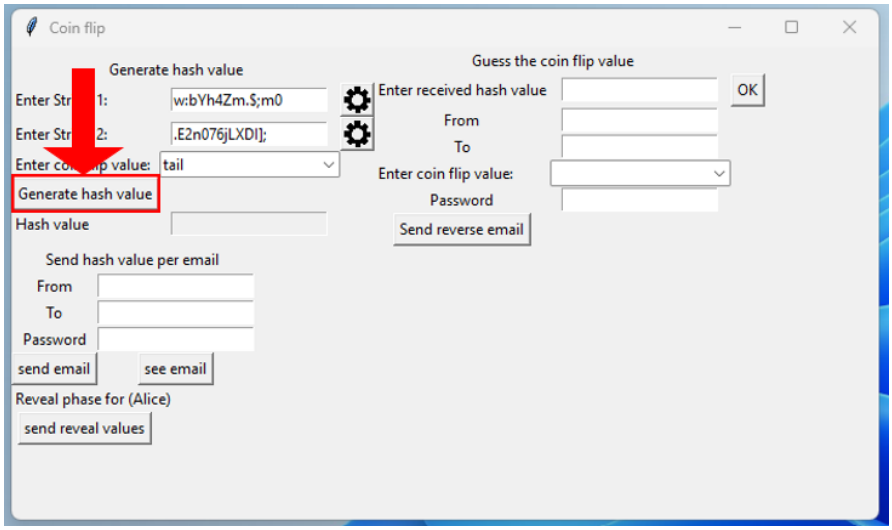

Figure 4: Generate hash value

Now a hash value has been generated and the values can be sent to Bob. For this, Alice needs to enter her email address in the "From:" field and Bob's email address in the "To:" field. Additionally, Alice must enter her password in the "Password" field. In this example, the email address a70904597@gmail.com was used for Alice, and p36494089@gmail.com was used for Bob. (Figure 5)

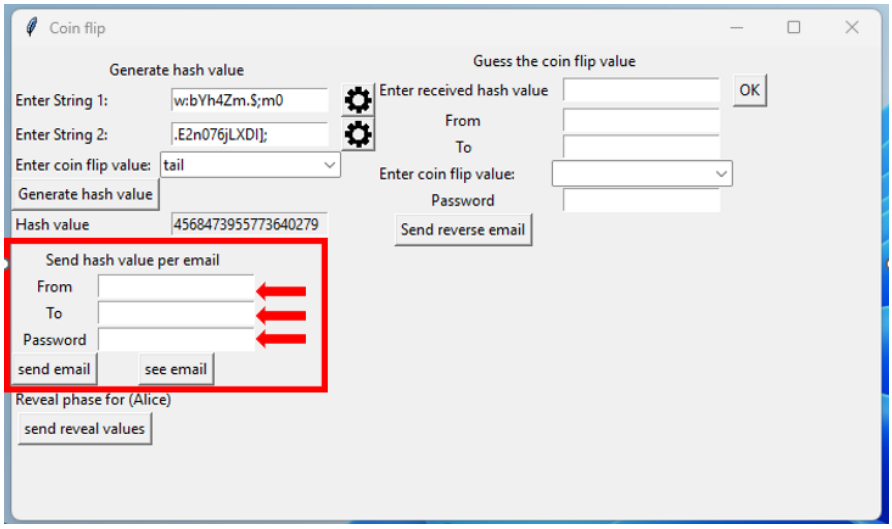

Figure 5: login credentials

After entering the credentials, the user has the option to view the draft of the email by clicking the 'see Email' button. There, the user can review all the content in a compact format. (See Figure 6)

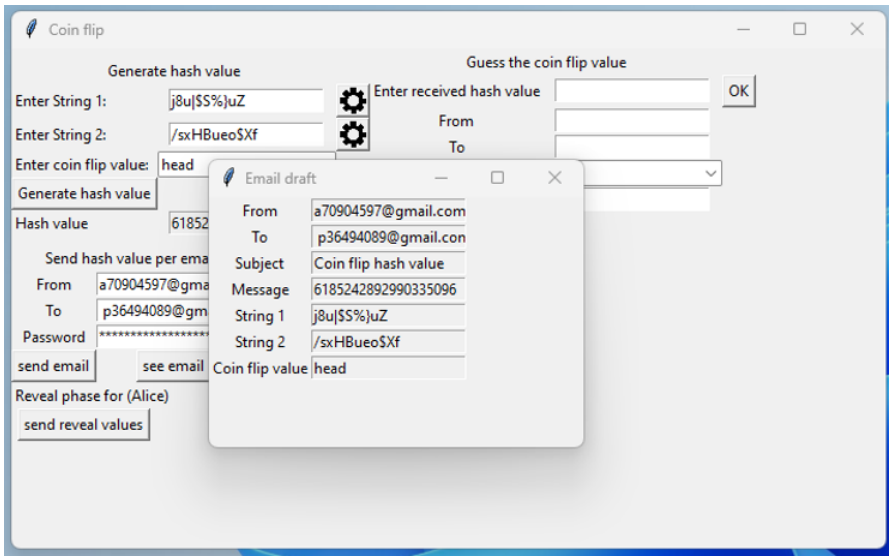

Figure 6: "see Mail" button

Now, the email can be sent by clicking on the "Send Mail" button. (Figure 7)

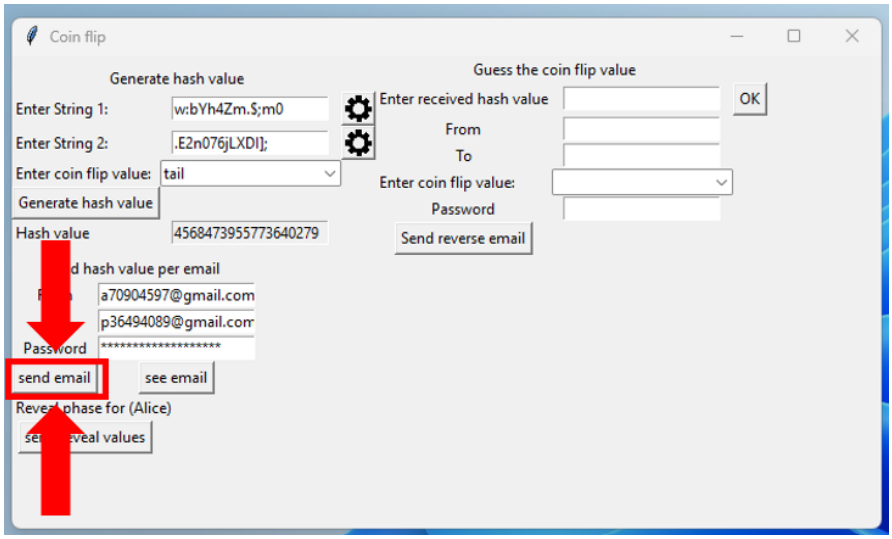

Figure 7: "Send Mail" button

Alice has now sent an email to Bob that contains both the first generated string and the hash value. This prevents Alice from altering her information without changing the hash value as well. Thus, the commit phase is complete, and in the next step, Bob can guess whether the coin has landed on heads or tails. Figure 7 shows a screenshot of the email. (Figure 8)

Figure 8: Email Hash Value

3.2 Phase 2 – Bob guesses the value of the coin

Now it's Bob's turn! After Bob has received the email from Alice containing String1 and the hash value, he can send his guess for the coin toss to Alice. He can start the application and simply enter the hash value he received from Alice into the designated field and click "ok". (Figure 9)

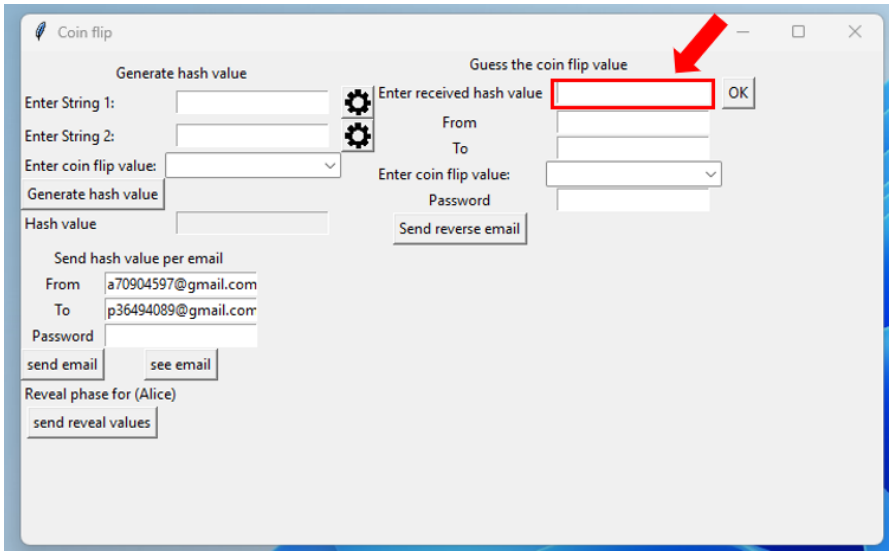

Figure 9: Enter the hash value

Afterwards, the email addresses will be automatically inserted into the fields, and Bob just needs to enter his password. Now Bob can choose between heads and tails, selecting his choice from the dropdown menu. (Figure 10)

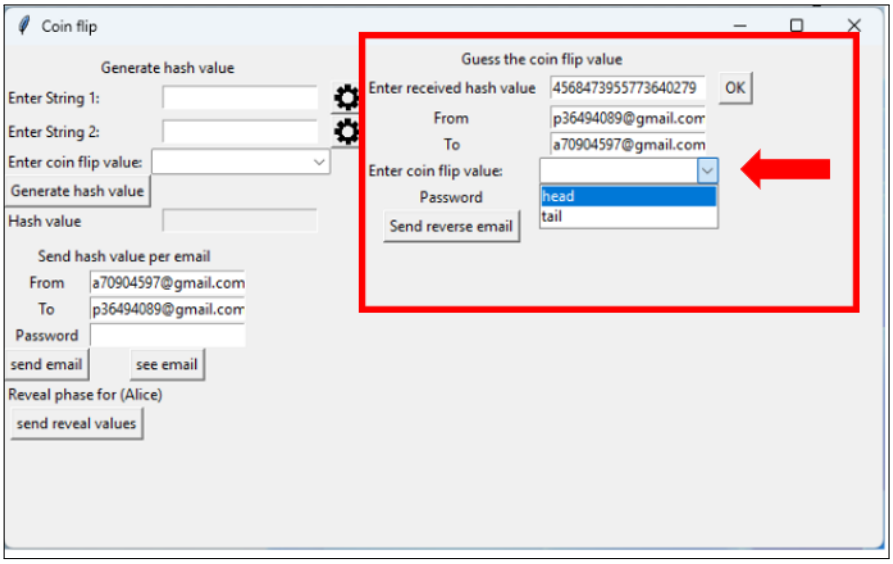

Figure 10: selecting choice from the dropdown menu

After Bob has made his choice, he can send the email and let Alice know his guess by clicking "Send Reverse mail". (Figure 11)

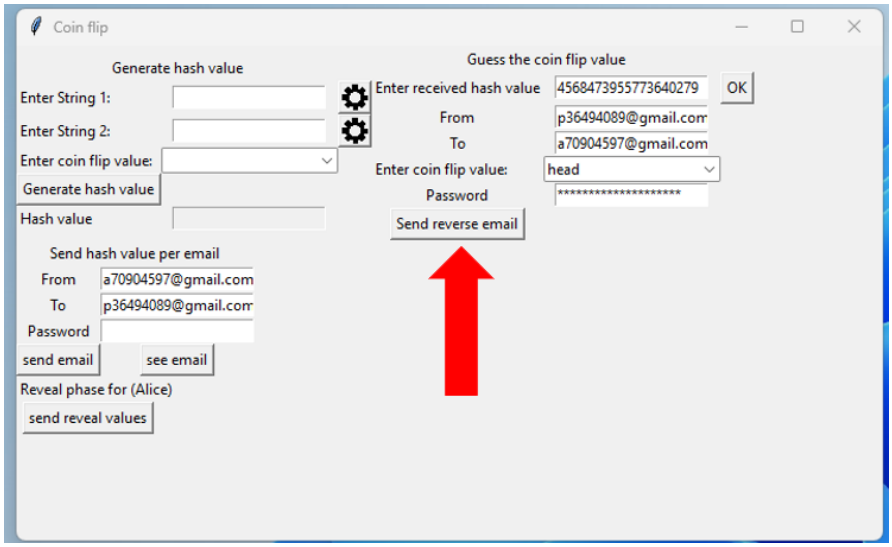

Figure 11: Send reverse mail

Alice has now received Bob's guess via email, as depicted in Figure 11. Bob can no longer change his guess, and Alice is ready to send the true result of the coin toss to Bob. And now the reveal phase can begin. (Figure 12)

Figure 12: Guessed Value

3.3 Phase 3 – Reveal phase and Bob's verification option

In the reveal phase, Alice can now inform Bob of the actual result of the coin toss by simply clicking the 'Send Reveal Values' button. (Figure 13)

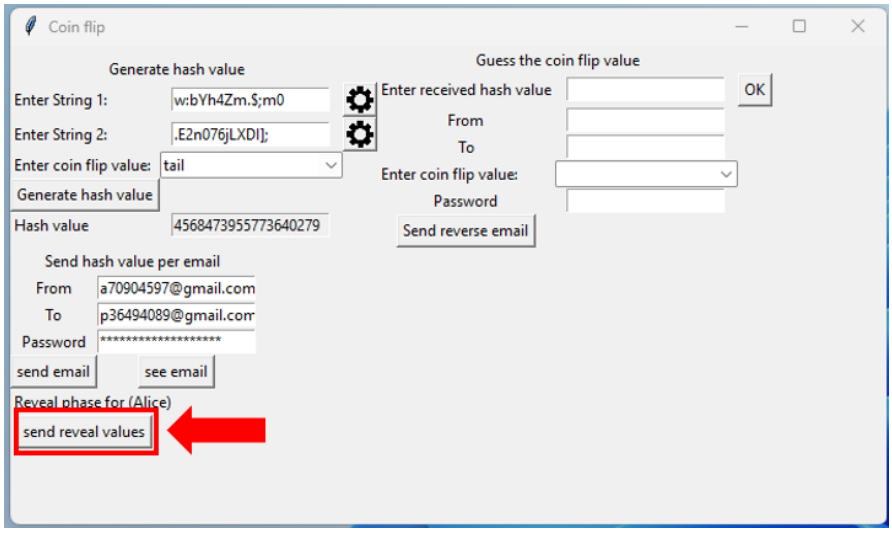

Figure 13: Send reveal values

The email consists of String 1, String 2, and the result of the coin toss. (Figure 14)

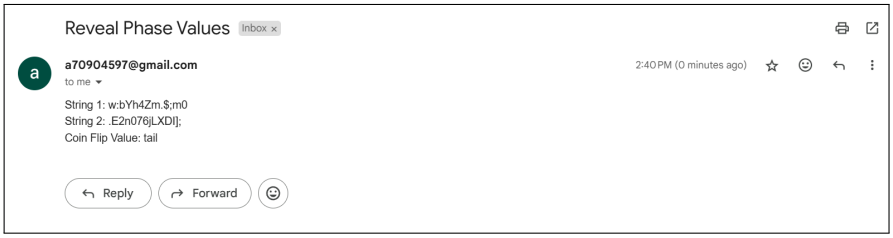

Figure 14: Reveal mail

If Bob wants to verify that this is indeed the result of the toss from Phase 1, he can simply enter the values received in the email into the corresponding fields and generate the hash value from them. If the hash values match, Bob can be sure that Alice was honest with him. (Figure 16)

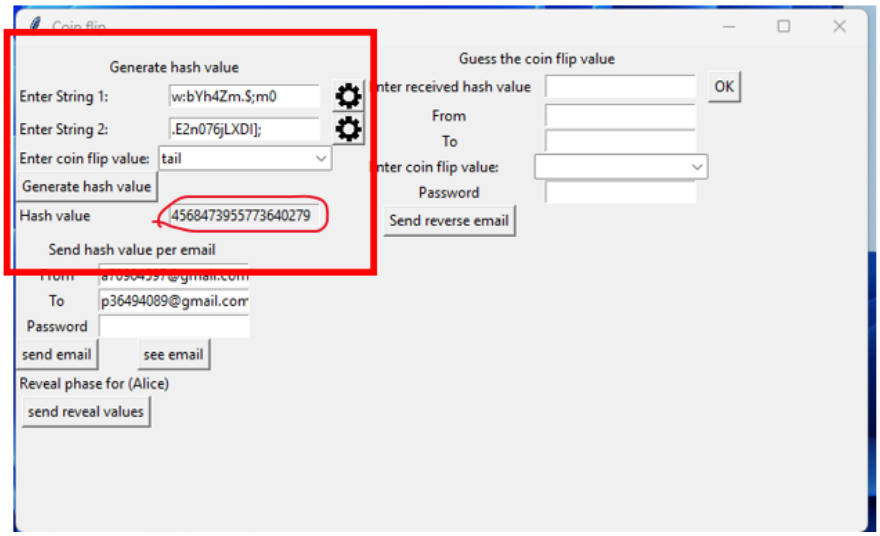

Figure 15: send reveal values

4 Logging

The application logs certain actions to make them more traceable. Each time an email is sent, information is recorded in the file 'log.txt'. The following details are saved:

Phase 1	**Phase 2**	**Phase 3**
Sender's name	Sender's name	Sender's name (2)
Sending date	Sending date	Sending date
Sender's email	Sender's email	Sender's email
Recipient's email	Recipient's email	Recipient's email
String 1	Guessed Value	String 1
Hash value		String 2
		Coin flip Value

Table 1: Logged data

In the following figure, the logging process is illustrated as an example.(Figure 16)

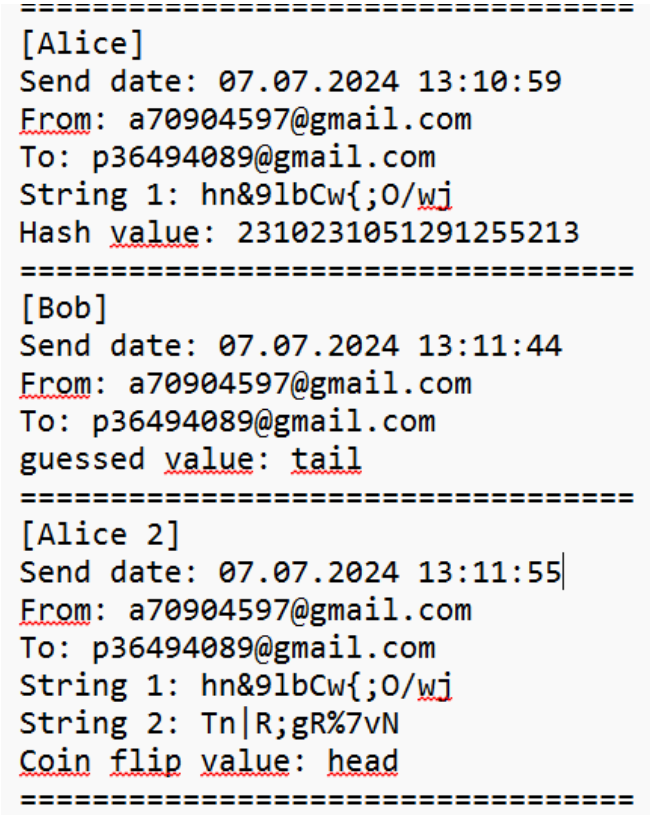

Figure 16: Logging

References

[Cap23] Clemens H. Cap. Zero knowledge proofs. Electronic document, 7 2023.

[CLRS09] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. *Introduction to Algorithms*. MIT Press, 3rd edition, 2009.