Hyoin Lee (20102711)

Step 1 – Knowledge Question (40-70 words) In your own words, describe what hashing is in general.

I would say hashing is like a process of key-to-address transformation. Unlike other searching methods which compare key values for retrieval, this method involves mapping the key value directly to the address of a memory location through a specific operation. Hashing reduces the need to compare key values when searching for records and allows for finding the desired record with just a single access. However, it requires converting all key values into numeric form, finding an appropriate hashing function, and addressing the issue of collision addresses generated by the calculation.

Step 2 – Knowledge Question (60-100 words) Research hashing algorithms. Describe advantages and disadvantages for at least three different hashing algorithms. Please provide references for external resources.

Hashing allows for fast data retrieval by mapping a key (or data) to a unique hash value, which serve as an index.

Hash functions are crucial in cryptography as they provide security by transforming data into a fixed-size hash that is difficult to reverse engineer.

Hashing allows us to detect if data has been altered or corrupted during transmission by comparing the hash value of received data with original hash value.

Despite good hash functions aim to minimize collisions, they are still possible and could occur security implications and impacts the efficiency of hash tables.

Hash functions are designed to be one-way functions, meaning that it’s computationally infeasible to reverse the process and obtain the original input data.

Hash function have a limited output range (fixed length), which means that no matter how large the input data is, the hash value will always be of a fixed size. This can lead to hash collisions in situations with many possible inputs.

Reference

<https://medium.com/@tutorlix69/what-are-the-advantages-and-disadvantages-of-hashing-8ba6fd8fadef>

Step 3 – Knowledge Question (50-90 words) Provide a stepwise description (algorithmic) of a) how you can store a password safely using hashing techniques and b) how you can verify that some string is the right password?

1. 1. Obtain a plaintext password

2. Apply hash algorithm to the plaintext password.

3. Store the hashed output in a database instead of the plaintext password.

1. 1. Retrieve the hashed password from the database based on the username or id

2. Obtain the password entered by the user for verification.

3. Hash the entered password using the same hash algorithm used for storage.

4. Compare the result of hash user entered with already stored.

5. If the hashes match, the entered password is correct, otherwise not correct.

Step 4 – Knowledge Question (20-40 words) What is the purpose of a “salt” when hashing a password? What are the two most important properties of a “salt”?

Salt is a randomly generated string of characters that is added to a password before hashing. This is to enhance the security of the hashed password.

The two most important features of salt are ‘uniqueness’ and ‘randomness’. With these features, the security of hashed password is significantly strengthened.

Step 5 – Add password to Player class.

1. Type “pip install argon2-cffi” to install the package on the command prompt and then check if the package is installed correctly by typing “pip show argon2-cffi”.
2. Argon2-cffi package offers hash and verify functions using argon2 algorithm to enhance the security of passwords that users provided in a convenient and secure way.