Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

Task 0 Execution:

Note: I get the function name from the github description Block JavaDoc and BlockChain Javadoc, following the description to build up the method.

Task 0 Block.java

```
package ds;
import com.google.gson.Gson;
import com.google.gson.GsonBuilder;
import com.google.gson.JsonObject;
import java.math.BigInteger;
import java.security.MessageDigest;
import java.security.NoSuchAlgorithmException;
import java.sql.Timestamp;

public class Block {
    //required variables for Task 0
    //Block(int index, java.sql.Timestamp timestamp, java.lang.String
data, int difficulty)
    private int index;
    private Timestamp timestamp;
    private String data;
    private String previousHash;
    private BigInteger nonce;
    private int difficulty;

    //Constructor
    public Block(int index, Timestamp timestamp, String data, int
difficulty) {
        this.index = index;
        this.timestamp = timestamp;
        this.data = data;
        this.difficulty = difficulty;
        this.previousHash = "";
        //initialize nounce
        this.nonce = BigInteger.ZERO;
    }
```

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

```
    //This method computes a hash of the concatenation of the index,
timestamp, data, previousHash, nonce, and difficulty.
    public String calculateHash() {
        try {
            //Use SHA-256 to compute the hash
            MessageDigest digest = MessageDigest.getInstance("SHA-
256");
            //String of the index, timestamp, data, previousHash,
nonce, and difficulty
            String input = index + timestamp.toString() + data +
previousHash + nonce.toString() + difficulty;
            //get the hashBytes
            byte[] hashBytes = digest.digest(input.getBytes());
            StringBuilder hexString = new StringBuilder();
            //Hexadecimal Conversion
            for (byte b : hashBytes) {
                String hex = Integer.toHexString(0xff & b);
                if (hex.length() == 1) hexString.append('0');
                hexString.append(hex);
            }
            return hexString.toString();
        } catch (NoSuchAlgorithmException e) {
            throw new RuntimeException(e);
        }
    }
    //The proof of work methods finds a good hash. It increments the
nonce until it produces a good hash.
    public String proofOfWork() {
        String target = "0".repeat(difficulty);
        //If the hash has the appropriate number of leading hex
zeroes, it is done and returns that proper hash.
        // If the hash does not have the appropriate number of leading
hex zeroes, it increments the nonce by 1 and tries again.
        // It continues this process, burning electricity and CPU
cycles, until it gets lucky and finds a good hash.
        while (!calculateHash().substring(0,
difficulty).equals(target)) {
```

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

```
            nonce = nonce.add(BigInteger.ONE);
        }
        return calculateHash();
    }
    //getter & setter
    public int getIndex() {
        return index;
    }

    public Timestamp getTimestamp() {
        return timestamp;
    }

    public String getData() {
        return data;
    }

    public String getPreviousHash() {
        return previousHash;
    }

    public BigInteger getNonce() {
        return nonce;
    }

    public int getDifficulty() {
        return difficulty;
    }

    public void setData(String data) {
        this.data = data;
    }

    public void setPreviousHash(String previousHash) {
        this.previousHash = previousHash;
    }
```

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

```
    public void setTimestamp(Timestamp timestamp) {
        this.timestamp = timestamp;
    }
    //toString method: set up the output by gson
    @Override
    public String toString() {
        Gson gson = new GsonBuilder().setPrettyPrinting().create();
        JsonObject blockObj = new JsonObject();
        blockObj.addProperty("index", getIndex());
        blockObj.addProperty("time stamp ",
getTimestamp().toString());
        blockObj.addProperty("Tx ", getData());
        blockObj.addProperty("PrevHash", getPreviousHash());
        blockObj.addProperty("nonce", getNonce().toString());
        blockObj.addProperty("difficulty", getDifficulty());

        return gson.toJson(blockObj);
    }
}
```

Task 0 BlockChain.java

```
package ds;
import com.google.gson.*;

import java.sql.Timestamp;
import java.util.ArrayList;
import java.util.Scanner;

public class BlockChain {
    //required variables
    private ArrayList<Block> chain;
    private String chainHash;
    private int hashesPerSecond;

    //default Constructor: initialize the block chain
```

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

```
    public BlockChain() {
        this.chain = new ArrayList<>();
        this.chainHash = "";
        Block genesis = new Block(0, getTime(), "Genesis", 2);
        genesis.proofOfWork();
        chain.add(genesis);
        this.chainHash = genesis.calculateHash();
    }
    //getter & setter
    public String getChainHash() {
        return chainHash;
    }

    public Timestamp getTime() {
        return new Timestamp(System.currentTimeMillis());
    }

    public Block getLatestBlock() {
        return chain.get(chain.size() - 1);
    }

    public int getChainSize() {
        return chain.size();
    }

    public int getHashesPerSecond() {
        return hashesPerSecond;
    }

    public void addBlock(Block newBlock) {
        newBlock.setPreviousHash(this.chainHash);
        newBlock.proofOfWork();
        this.chainHash = newBlock.calculateHash();
        chain.add(newBlock);
    }

    public Block getBlock(int i) {
```

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

```
        return chain.get(i);
    }

    //Compute and return the total difficulty of all blocks on the
chain. Each block knows its own difficulty.
    public int getTotalDifficulty() {
        return chain.stream().mapToInt(Block::getDifficulty).sum();
    }

    //Compute and return the expected number of hashes required for
the entire chain.
    public double getTotalExpectedHashes() {
        return chain.stream().mapToDouble(b -> Math.pow(16,
b.getDifficulty())).sum();
    }

    //This method computes exactly 2 million hashes and times how
long that process takes.
    // So, hashes per second is approximated as (2 million / number
of seconds).
    // It is run on start up and sets the instance variable
hashesPerSecond.
    // It uses a simple string - "00000000" to hash.
    public void computeHashesPerSecond() {
        long start = System.currentTimeMillis();
        for (int i = 0; i < 2000000; i++) {
            new Block(0, getTime(), "test", 2).calculateHash();
        }
        long end = System.currentTimeMillis();
        hashesPerSecond = (int) (2000000 / (end - start) * 1000);
    }
    //If the chain only contains one block, the genesis block at
position 0,
    // this routine computes the hash of the block and checks that
the hash has the requisite number of leftmost 0's (proof of work)
    // as specified in the difficulty field. It also checks that the
chain hash is equal to this computed hash.
```

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

```
    // If either check fails, return an error message. Otherwise,
return the string "TRUE".
    // If the chain has more blocks than one, begin checking from
block one. Continue checking until you have validated the entire
chain.
    // The first check will involve a computation of a hash in Block
0 and a comparison with the hash pointer in Block 1.
    // If they match and if the proof of work is correct, go and
visit the next block in the chain.
    // At the end, check that the chain hash is also correct.
    public String isChainValid() {
        for (int i = 1; i < chain.size(); i++) {
            if (!chain.get(i).getPreviousHash().equals(chain.get(i -
1).calculateHash())) {
                return "FALSE";
            }
        }
        return "TRUE";
    }
    //Check which block is invalid
    public int inValidBlock(){
        for (int i = 1; i < chain.size(); i++) {
            if (!chain.get(i).getPreviousHash().equals(chain.get(i -
1).calculateHash())) {
                return i-1;
            }
        }
        return -1;
    }
    //This routine repairs the chain.
    // It checks the hashes of each block and ensures that any
illegal hashes are recomputed.
    // After this routine is run, the chain will be valid.
    // The routine does not modify any difficulty values.
    // It computes new proof of work based on the difficulty
specified in the Block.
    public void repairChain() {
```

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

```
        for (int i = 1; i < chain.size(); i++) {
            chain.get(i).proofOfWork();
            chain.get(i).setPreviousHash(chain.get(i -
1).calculateHash());
        }
        this.chainHash = getLatestBlock().calculateHash();
    }

    //toString method: set up the block information by Block.toString
    //Finally add the chainHash in the last part
    @Override
    public String toString() {
        StringBuilder sb = new StringBuilder();
        Gson gson = new GsonBuilder().setPrettyPrinting().create();
        JsonObject blockchainObj = new JsonObject();

        JsonArray dsChainArray = new JsonArray();

        //Use StringBuilder to add the Json info
        for (Block block : chain) {
            dsChainArray.add(new JsonParser().parse(block.toString()));
        }

        //add the ds_chain and chainHash
        blockchainObj.add("ds_chain", dsChainArray);
        blockchainObj.addProperty("chainHash", this.chainHash);


        return gson.toJson(blockchainObj);
    }


    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        //create a blockchain
        BlockChain blockchain = new BlockChain();
        //get the computeHashesPerSecond
```

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

```
        blockchain.computeHashesPerSecond();



        //Loop the selection manual
        while (true) {
            System.out.println("0. View basic blockchain status.");
            System.out.println("1. Add a transaction to the
blockchain.");
            System.out.println("2. Verify the blockchain.");
            System.out.println("3. View the blockchain.");
            System.out.println("4. Corrupt the chain.");
            System.out.println("5. Hide the corruption by repairing the
chain.");
            System.out.println("6. Exit.");
            int choice = scanner.nextInt();
            scanner.nextLine();

            switch (choice) {
                //0. View basic blockchain status.
                case 0:
                    System.out.println("Current size of chain: " +
blockchain.getChainSize());
                    System.out.println("Difficulty of most recent block:
" + blockchain.getLatestBlock().getDifficulty());
                    System.out.println("Total difficulty for all blocks:
" + blockchain.getTotalDifficulty());
                    System.out.println("Experimented with 2,000,000
hashes.");
                    System.out.println("Approximate hashes per second: "
+ blockchain.getHashesPerSecond());
                    System.out.println("Expected total hashes required:
" + blockchain.getTotalExpectedHashes());
                    System.out.println("Nonce for most recent block: " +
blockchain.getLatestBlock().getNonce());
                    System.out.println("Chain hash: " +
blockchain.getChainHash());
                    break;
```

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

```
            // 1. Add a transaction to the blockchain.
            case 1:
                System.out.print("Enter difficulty > 1: ");
                int difficulty = scanner.nextInt();
                scanner.nextLine();
                System.out.print("Enter transaction: ");
                String data = scanner.nextLine();
                long startTime = System.currentTimeMillis();
                blockchain.addBlock(new
Block(blockchain.getChainSize(), blockchain.getTime(), data,
difficulty));
                long endTime = System.currentTimeMillis();
                System.out.println("Total execution time to add this
block was " + (endTime - startTime) + " milliseconds");
                break;
            // 2. Verify the blockchain.
            case 2:
                if(blockchain.isChainValid().equals("TRUE")){
                    long verifyStartTime =
System.currentTimeMillis();
                    System.out.println("Verifying entire chain");
                    System.out.println("Chain verification: " +
blockchain.isChainValid());
                    long verifyEndTime = System.currentTimeMillis();
                    long totalVerificationTime = verifyEndTime -
verifyStartTime;
                    System.out.println("Total execution time required
to verify the chain was "+ totalVerificationTime +" milliseconds");
                }else{
                    System.out.println("Chain verification failed");
                    System.out.println("Verifying entire chain");
                    System.out.println("Improper hash on node " +
blockchain.inValidBlock() + " Does not begin with 0000");
                }
                break;
            // 3. View the blockchain.
            case 3:
```

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

```
                System.out.println("View the Blockchain");
                System.out.println(blockchain.toString());
                break;
            //4. Corrupt the chain.
            case 4:
                System.out.print("Enter block ID to corrupt: ");
                int blockId = scanner.nextInt();
                scanner.nextLine();
                System.out.print("Enter new data: ");
                String newData = scanner.nextLine();
                blockchain.getBlock(blockId).setData(newData);
                System.out.println("Block " + blockId + " now holds:
" + newData);
                break;
            // 5. Hide the corruption by repairing the chain.
            case 5:
                long repairStart = System.currentTimeMillis();
                blockchain.repairChain();
                long repairEnd = System.currentTimeMillis();
                System.out.println("Total execution time required to
repair the chain was " + (repairEnd - repairStart) + "
milliseconds");
                break;
            case 6:
                scanner.close();
                return;
            default:
                System.out.println("Invalid choice. Try again.");
        }
    }
}
```

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

Output:

C:\Users\USER\.jdks\openjdk-22.0.2\bin\java.exe "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA 2024.2.0.2\lib\idea_rt.jar=53822:C:\Program Files\JetBrains\IntelliJ IDEA 2024.2.0.2\bin" -Dfile.encoding=UTF-8 -Dsun.stdout.encoding=UTF-8 -Dsun.stderr.encoding=UTF-8 -classpath C:\Users\USER\IdeaProjects\Project3Task0\target\classes;C:\Users\USER\.m2\repository\com\google\code\gson\gson\2.9.0\gson-2.9.0.jar ds.BlockChain

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

0

Current size of chain: 1

Difficulty of most recent block: 2

Total difficulty for all blocks: 2

Experimented with 2,000,000 hashes.

Approximate hashes per second: 666000

Expected total hashes required: 256.0

Nonce for most recent block: 724

Chain hash:
007211febeb572d4ae6edab3036b52d2a6026fb2c43feb7d58cb92347b68a6c1

0. View basic blockchain status.

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

1

Enter difficulty > 1: 4

Enter transaction: Alice pays Bob 100 DSCoin

Total execution time to add this block was 120 milliseconds

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

1

Enter difficulty > 1: 4

Enter transaction: Bob pays Carol 20 DSCoin

Total execution time to add this block was 29 milliseconds

0. View basic blockchain status.

1. Add a transaction to the blockchain.

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

1

Enter difficulty > 1: 4

Enter transaction: Carol pays Donna 10 DSCoin

Total execution time to add this block was 63 milliseconds

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

3

View the Blockchain


```
{
   "ds_chain": [
      {
         "index": 0,
         "time stamp ": "2025-03-18 02:54:31.837",
```

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

"Tx ": "Genesis",

"PrevHash": "",

"nonce": "724",

"difficulty": 2

},

{

"index": 1,

"time stamp ": "2025-03-18 02:54:55.184",

"Tx ": "Alice pays Bob 100 DSCoin",

"PrevHash":
"007211febeb572d4ae6edab3036b52d2a6026fb2c43feb7d58cb92347b68a6c1",

"nonce": "39277",

"difficulty": 4

},

{

"index": 2,

"time stamp ": "2025-03-18 02:55:02.597",

"Tx ": "Bob pays Carol 20 DSCoin",

"PrevHash":
"0000307e7793879209cb30d4c981ece72d958d6c558e55b47f6fe42308ac8b90",

"nonce": "24626",

"difficulty": 4

},

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

{

"index": 3,

"time stamp ": "2025-03-18 02:55:08.264",

"Tx ": "Carol pays Donna 10 DSCoin",

"PrevHash": "000073ccafc3675abc16f28b412c3c7c9cede312e44773fe8170525a989ca396",

"nonce": "27808",

"difficulty": 4

}

],

"chainHash": "0000da7ff0bcaf8d46f162d2f2b6ff4cc66ca07030b0d4ab7ab6b80698fd3d21"

}

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

2

Verifying entire chain

Chain verification: TRUE

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

Total execution time required to verify the chain was 0 milliseconds

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

4

Enter block ID to corrupt: 2

Enter new data: Block 2 now holds Bob pays Tony 30 DSCoin

Block 2 now holds: Block 2 now holds Bob pays Tony 30 DSCoin

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

3

View the Blockchain

{

"ds_chain": [

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

{

"index": 0,

"time stamp ": "2025-03-18 02:54:31.837",

"Tx ": "Genesis",

"PrevHash": "",

"nonce": "724",

"difficulty": 2

},

{

"index": 1,

"time stamp ": "2025-03-18 02:54:55.184",

"Tx ": "Alice pays Bob 100 DSCoin",

"PrevHash":
"007211febeb572d4ae6edab3036b52d2a6026fb2c43feb7d58cb92347b68a6c1",

"nonce": "39277",

"difficulty": 4

},

{

"index": 2,

"time stamp ": "2025-03-18 02:55:02.597",

"Tx ": "Block 2 now holds Bob pays Tony 30 DSCoin",

"PrevHash":
"0000307e7793879209cb30d4c981ece72d958d6c558e55b47f6fe42308ac8b90",

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

"nonce": "24626",

"difficulty": 4

},

{

"index": 3,

"time stamp ": "2025-03-18 02:55:08.264",

"Tx ": "Carol pays Donna 10 DSCoin",

"PrevHash": "000073ccafc3675abc16f28b412c3c7c9cede312e44773fe8170525a989ca396",

"nonce": "27808",

"difficulty": 4

}

],

"chainHash": "0000da7ff0bcaf8d46f162d2f2b6ff4cc66ca07030b0d4ab7ab6b80698fd3d21"

}

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

2

Chain verification failed

Verifying entire chain

Improper hash on node 2 Does not begin with 0000

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

5

Total execution time required to repair the chain was 33 milliseconds

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

2

Verifying entire chain

Chain verification: TRUE

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

Total execution time required to verify the chain was 0 milliseconds

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

3

View the Blockchain


```
{
   "ds_chain": [
      {
         "index": 0,
         "time stamp ": "2025-03-18 02:54:31.837",
         "Tx ": "Genesis",
         "PrevHash": "",
         "nonce": "724",
         "difficulty": 2
      },
      {
         "index": 1,
         "time stamp ": "2025-03-18 02:54:55.184",
```

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

"Tx ": "Alice pays Bob 100 DSCoin",

"PrevHash": "007211febeb572d4ae6edab3036b52d2a6026fb2c43feb7d58cb92347b68a6c1",

"nonce": "39277",

"difficulty": 4

},

{

"index": 2,

"time stamp ": "2025-03-18 02:55:02.597",

"Tx ": "Block 2 now holds Bob pays Tony 30 DSCoin",

"PrevHash": "0000307e7793879209cb30d4c981ece72d958d6c558e55b47f6fe42308ac8b90",

"nonce": "48735",

"difficulty": 4

},

{

"index": 3,

"time stamp ": "2025-03-18 02:55:08.264",

"Tx ": "Carol pays Donna 10 DSCoin",

"PrevHash": "00007e37699939c5b584be312dad8c4352e8163f33f1c32dd114dd3228bba120",

"nonce": "27808",

"difficulty": 4

}

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

],

"chainHash":
"ea1811dc6fff7127cf92b20c5b2b74e380b8ea7ea3819dd4682909c719236b47"

}

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

6

Process finished with exit code 0

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

Task 1

ClientTCP.java

```
package ds;

import com.google.gson.Gson;

import java.net.*;
import java.io.*;
import java.util.Scanner;

public class ClientTCP {

    public static void main(String args[]) {
        // arguments supply hostname
        Socket clientSocket = null;
        //Scanner for client input
        Scanner scanner = new Scanner(System.in);
        try {
            //serverPort = 7777
            int serverPort = 7777;
            //connect to port 7777
            clientSocket = new Socket( "localhost", serverPort);
            System.out.println("Client is running...");
            //use BufferReader to read the response
            BufferedReader in = new BufferedReader(new
InputStreamReader(clientSocket.getInputStream()));
            //use PrintWriter to send the request
            PrintWriter out = new PrintWriter(new BufferedWriter(new
OutputStreamWriter(clientSocket.getOutputStream())));
            //gson for request message
            Gson gson = new Gson();

            while (true) {
                //Selection manual
                System.out.println("0. View basic blockchain status.");
```

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

```
            System.out.println("1. Add a transaction to the
blockchain.");
            System.out.println("2. Verify the blockchain.");
            System.out.println("3. View the blockchain.");
            System.out.println("4. Corrupt the chain.");
            System.out.println("5. Hide the corruption by repairing
the chain.");
            System.out.println("6. Exit.");
            //input the selection
            int choice = scanner.nextInt();
            scanner.nextLine();



            if (choice == 0) {
                //0. View basic blockchain status.
                //set the request message
                RequestMessage requestMessage = new
RequestMessage("status");
                //set into json
                String jsonRequest = gson.toJson(requestMessage);
                //send the request and use flush() to confirm the
streamline
                out.println(jsonRequest);
                out.flush();
                //received the server response
                String responseData = in.readLine();
                if (responseData != null) {
                    //get the response message
                    ResponseMessage responseMessage =
gson.fromJson(responseData, ResponseMessage.class);
                    //check if server processed the request
                    if
("success".equals(responseMessage.getStatus())) {
                        // Deserialize JSON response to an
appropriate object
                        BlockChain status =
gson.fromJson(responseMessage.getMessage(), BlockChain.class);
```

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

```
                        // Display blockchain status
                        System.out.println("Current size of chain: "
+ status.getChainSize());
                        System.out.println("Difficulty of most recent
block: " + status.getLatestBlock().getDifficulty());
                        System.out.println("Total difficulty for all
blocks: " + status.getTotalDifficulty());
                        System.out.println("Experimented with
2,000,000 hashes.");
                        System.out.println("Approximate hashes per
second: " + status.getHashesPerSecond());
                        System.out.println("Expected total hashes
required: " + status.getTotalExpectedHashes());
                        System.out.println("Nonce for most recent
block: " + status.getLatestBlock().getNonce());
                        System.out.println("Chain hash: " +
status.getChainHash());
                    } else {
                        System.out.println("Error: " +
responseMessage.getMessage());
                    }

                } else {
                    System.out.println("Error: No response received
from the server.");
                }

            }else if (choice == 1){
                // "1. Add a transaction to the blockchain."
                // get the startTime
                long startTime = System.currentTimeMillis();
                //set difficulty
                System.out.print("Enter difficulty > 1: ");
                int difficulty = scanner.nextInt();
                scanner.nextLine();
                //set the transaction
```

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

```
                System.out.print("Enter transaction: ");
                String data = scanner.nextLine();

                //set the requestMessage and sent to the server
                RequestMessage requestMessage = new
RequestMessage("add transaction", data, difficulty);
                String jsonRequest = gson.toJson(requestMessage);
                //do the same thing as previous option
                out.println(jsonRequest);
                out.flush();
                //receive the server response
                String responseData = in.readLine();
                //check if the response has value
                if (responseData != null) {
                    //get the responseMessage
                    ResponseMessage responseMessage =
gson.fromJson(responseData, ResponseMessage.class);
                    //Check if server processed the request
                    if
("success".equals(responseMessage.getStatus())) {
                        // Deserialize JSON response to an
appropriate object
                        BlockChain status =
gson.fromJson(responseData, BlockChain.class);
                        long endTime = System.currentTimeMillis();
                        //calculate the processing time
                        System.out.println("Total execution time to
add this block was " + (endTime - startTime) + " milliseconds");
                    } else {
                        System.out.println("Error: " +
responseMessage.getMessage());
                    }

                } else {
                    System.out.println("Error: No response received
from the server.");
                }
```

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

```
            }else if (choice == 2){
                // 2. Verify the blockchain.
                // same thing as previous selection
                long verifyStartTime = System.currentTimeMillis();
                RequestMessage requestMessage = new
RequestMessage("verify");
                String jsonRequest = gson.toJson(requestMessage);

                out.println(jsonRequest);
                out.flush();

                String responseData = in.readLine();

                if (responseData != null) {

                    ResponseMessage responseMessage =
gson.fromJson(responseData, ResponseMessage.class);
                    if
("success".equals(responseMessage.getStatus())) {

if(responseMessage.getMessage().equals("TRUE")){
                        //check the chain verification
                        System.out.println("Verifying entire
chain");
                        System.out.println("Chain verification: "
+ responseMessage.getMessage());
                        long verifyEndTime =
System.currentTimeMillis();
                        //get the total verification time.
                        long totalVerificationTime = verifyEndTime
- verifyStartTime;
                        System.out.println("Total execution time
required to verify the chain was "+ totalVerificationTime +"
milliseconds");
                    }else{
                        System.out.println("Verifying entire
```

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

```
chain");
                            System.out.println("Chain verification: "
+ responseMessage.getMessage());
                            System.out.println("Improper hash on node
" + responseMessage.getInvalidBlock() + " Does not begin with 0000");
                        }

                    } else {
                        System.out.println("Error: " +
responseMessage.getMessage());
                    }


                } else {
                    System.out.println("Error: No response received
from the server.");

                }


            }else if (choice == 3){
                //3. View the blockchain.
                //do the same thing as previous request, but
different action message
                RequestMessage requestMessage = new
RequestMessage("view");
                String jsonRequest = gson.toJson(requestMessage);

                out.println(jsonRequest);
                out.flush();

                String responseData = in.readLine();
                if (responseData != null) {
                    ResponseMessage responseMessage =
gson.fromJson(responseData, ResponseMessage.class);
                    if
("success".equals(responseMessage.getStatus())) {
```

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

```
                        BlockChain status =
gson.fromJson(responseMessage.getMessage(), BlockChain.class);
                        System.out.println("View the Blockchain");
                        //print out the whole blockchain
                        System.out.println(status.toString());
                    } else {
                        System.out.println("Error: " +
responseMessage.getMessage());
                    }
                } else {
                    System.out.println("Error: No response received
from the server.");
                }

            }else if (choice == 4){
                //4. Corrupt the chain.
                //do the same thing as previous request, but
different action message
                //set the corrupt id
                System.out.print("Enter block ID of block to
corrupt: ");
                int blockId = scanner.nextInt();
                scanner.nextLine();
                //enter the new data
                System.out.print("Enter new data for block "+
blockId+ ": ");
                String newData = scanner.nextLine();
                //send the request to server
                RequestMessage requestMessage = new
RequestMessage("corrupt", blockId, newData);
                String jsonRequest = gson.toJson(requestMessage);

                out.println(jsonRequest);
                out.flush();

                String responseData = in.readLine();
```

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

```
                if (responseData != null) {
                    ResponseMessage responseMessage =
gson.fromJson(responseData, ResponseMessage.class);
                    if
("success".equals(responseMessage.getStatus())) {
                        //print out the block that user intend to
change
                        System.out.println("Block " + blockId + " now
holds: " + newData);
                    } else {
                        System.out.println("Error: " +
responseMessage.getMessage());
                    }

                } else {
                    System.out.println("Error: No response received
from the server.");
                }

            }else if (choice == 5){
                //5. Hide the corruption by repairing the chain.
                //do the same thing as previous selection, but
different action message
                long repairStart = System.currentTimeMillis();
                RequestMessage requestMessage = new
RequestMessage("hide");
                String jsonRequest = gson.toJson(requestMessage);

                out.println(jsonRequest);
                out.flush();

                String responseData = in.readLine();

                if (responseData != null) {
                    ResponseMessage responseMessage =
gson.fromJson(responseData, ResponseMessage.class);
                    if
```

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

```
("success".equals(responseMessage.getStatus())) {
                        long repairEnd = System.currentTimeMillis();
                        System.out.println("Repairing the entire
chain");
                        //print out the total repaired time
                        System.out.println("Total execution time
required to repair the chain was " + (repairEnd - repairStart) + "
milliseconds");

                    } else {
                        System.out.println("Error: " +
responseMessage.getMessage());
                    }

                } else {
                    System.out.println("Error: No response received
from the server.");
                }

            }else if (choice == 6){
                //6. Exit.
                scanner.close();
                return;
            }
        }

    } catch (IOException e) {
        System.out.println("IO Exception:" + e.getMessage());
    } finally {
        try {
            if (clientSocket != null) {
                clientSocket.close();
            }
        } catch (IOException e) {
            // ignore exception on close
        }
    }
```

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

```
    }
}
//request from client
class RequestMessage {
    //request: request action
    private String request;
    //info that need client to set
    private String data;
    private int difficulty;
    private int blockID;
    //Constructors
    public RequestMessage(String request, String data, int
difficulty) {
        this.request = request;
        this.data = data;
        this.difficulty = difficulty;
    }

    public RequestMessage(String request) {
        this.request = request;
    }

    public RequestMessage(String request, int blockID, String data) {
        this.request = request;
        this.data = data;
        this.blockID = blockID;
    }
    //getters
    public String getRequest() { return request; }
    public String getData() { return data; }
    public int getDifficulty() { return difficulty; }
    public int getBlockID() { return blockID; }
}

class ResponseMessage {
    //status: to confirm server process the request or not
    private String status;
```

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

```
    //info for making response
    private BlockChain blockChain;
    private String message;
    private int invalidBlock;
    //constructor
    public ResponseMessage(String status, BlockChain blockChain) {
        this.status = status;
        this.blockChain = blockChain;
    }

    public ResponseMessage(String status, String message) {
        this.status = status;
        this.message = message;
    }
    public ResponseMessage(String status, String message, int
invalidBlock) {
        this.status = status;
        this.message = message;
        this.invalidBlock = invalidBlock;
    }
    //getters
    public String getStatus() { return status; }
    public BlockChain getBlockChain() { return blockChain; }
    public String getMessage() { return message; }
    public int getInvalidBlock() { return invalidBlock; }
}
```

ServerTCP.java

```
package ds;

import com.google.gson.Gson;

import java.net.*;
import java.io.*;
import java.util.Scanner;

public class ServerTCP {
```

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

```
    public static void main(String args[]) {
        Socket clientSocket = null;
        BlockChain blockChain = new BlockChain();
        Gson gson = new Gson();

        try {
            int serverPort = 7777; // the server port we are using

            // Create a new server socket
            ServerSocket listenSocket = new ServerSocket(serverPort);
            System.out.println("Blockchain server is running");
            clientSocket = listenSocket.accept();
            /*
             * Block waiting for a new connection request from a
client.
             * When the request is received, "accept" it, and the rest
             * the tcp protocol handshake will then take place, making
             * the socket ready for reading and writing.
             */
            BufferedReader in = new BufferedReader(new
InputStreamReader(clientSocket.getInputStream()));
            PrintWriter out = new PrintWriter(new BufferedWriter(new
OutputStreamWriter(clientSocket.getOutputStream())), true);

            while (true) {
                //received the request from client
                String jsonRequest = in.readLine();
                if (jsonRequest == null) {
                    break;
                }
                //get the requestMessage, process the request by
handleRequest
                RequestMessage request = gson.fromJson(jsonRequest,
RequestMessage.class);
                ResponseMessage response = handleRequest(request,
blockChain);
```

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

```
            //get the response
            String jsonResponse = gson.toJson(response);
            //send back to client
            out.println(jsonResponse);
        }

        // Handle exceptions
    } catch (IOException e) {
        System.out.println("IO Exception:" + e.getMessage());

        // If quitting (typically by you sending quit signal) clean
up sockets
    } finally {
        try {
            if (clientSocket != null) {
                clientSocket.close();
            }
        } catch (IOException e) {
            // ignore exception on close
        }
    }
}
// handle the different request from client
private static ResponseMessage handleRequest(RequestMessage
request, BlockChain blockchain) {
    //gson from print out the request and response
    Gson gson = new Gson();
    String blockchainJson = gson.toJson(blockchain);
    String cleanMessage = "";
    System.out.println("We have a visitor");
    System.out.println("THE JSON REQUEST MESSAGE IS SHOWN HERE");
    // Print the request as a clean JSON string
    System.out.println(gson.toJson(request));
    System.out.println("THE JSON RESPONSE MESSAGE IS SHOWN HERE");

    ResponseMessage response;
    if(request.getRequest().equals("status")){
```

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

```
            response = new ResponseMessage("success", blockchainJson);
            System.out.println(gson.toJson(response));
            System.out.println("Number of Blocks on Chain == "+
blockchain.getChainSize());
        } else if (request.getRequest().equals("add transaction")) {
            //Selection 1
            //add the block to blockchain by client's input
            blockchain.addBlock(new Block(blockchain.getChainSize(),
blockchain.getTime(), request.getData(), request.getDifficulty()));
            response = new ResponseMessage("success", blockchain);
            System.out.println(gson.toJson(response));
            System.out.println("Number of Blocks on Chain == "+
blockchain.getChainSize());
        } else if (request.getRequest().equals("verify")) {
            //Selection 2
            //check the blockchain is valid or not
            if(blockchain.isChainValid().equals("TRUE")){
                response = new ResponseMessage("success",
blockchain.isChainValid());
                System.out.println(gson.toJson(response));
                System.out.println("Number of Blocks on Chain == "+
blockchain.getChainSize());
            }else{
                response = new ResponseMessage("success",
blockchain.isChainValid(), blockchain.inValidBlock());
                System.out.println(gson.toJson(response));
                System.out.println("Number of Blocks on Chain == "+
blockchain.getChainSize());
            }
        } else if (request.getRequest().equals("view")) {
            //Selection 3
            //print out the whole block in blockchain
            response = new ResponseMessage("success", blockchainJson);
            System.out.println(gson.toJson(response));
            System.out.println("Number of Blocks on Chain == "+
blockchain.getChainSize());
        } else if (request.getRequest().equals("corrupt")) {
```

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

```
            //Selection 4
            //corrupt the block by client's input

blockchain.getBlock(request.getBlockID()).setData(request.getData());
            response = new ResponseMessage("success", "Block " +
request.getBlockID() + " modified.");
            System.out.println(gson.toJson(response));
            System.out.println("Number of Blocks on Chain == "+
blockchain.getChainSize());
        } else if (request.getRequest().equals("hide")) {
            //Selection 5
            //repair the blockchain
            blockchain.repairChain();
            response = new ResponseMessage("success", "Blockchain
repaired.");
            System.out.println(gson.toJson(response));
            System.out.println("Number of Blocks on Chain == "+
blockchain.getChainSize());
        }else{
            response = new ResponseMessage("failed", "Invalid
request.");
            System.out.println(gson.toJson(response));
        }
        return response;

    }
}
```

Client Output:

C:\Users\USER\.jdks\openjdk-22.0.2\bin\java.exe "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA 2024.2.0.2\lib\idea_rt.jar=53790:C:\Program Files\JetBrains\IntelliJ IDEA 2024.2.0.2\bin" -Dfile.encoding=UTF-8 -Dsun.stdout.encoding=UTF-8 -Dsun.stderr.encoding=UTF-8 -classpath C:\Users\USER\IdeaProjects\Project3Task1\target\classes;C:\Users\USER\.m2\rep

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

ository\com\google\code\gson\gson\2.9.0\gson-2.9.0.jar ds.ClientTCP

Client is running...

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

0

Current size of chain: 1

Difficulty of most recent block: 2

Total difficulty for all blocks: 2

Experimented with 2,000,000 hashes.

Approximate hashes per second: 0

Expected total hashes required: 256.0

Nonce for most recent block: 379

Chain hash:
007edecc1e1adb97dd4da9bee199b573cfc4a53c8492607d484eb67ede6148a2

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

1

Enter difficulty > 1: 4

Enter transaction: Alice pays Bob 100 DSCoin

Total execution time to add this block was 21881 milliseconds

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

1

Enter difficulty > 1: 4

Enter transaction: Bob pays Carol 20 DSCoin

Total execution time to add this block was 5145 milliseconds

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

5. Hide the corruption by repairing the chain.

6. Exit.

1

Enter difficulty > 1: 4

Enter transaction: Carol pays Donna 10 DSCoin

Total execution time to add this block was 6789 milliseconds

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

3

View the Blockchain


```
{
   "ds_chain": [
      {
         "index": 0,
         "time stamp ": "2025-03-18 02:50:46.0",
         "Tx ": "Genesis",
         "PrevHash": "",
         "nonce": "379",
```

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

"difficulty": 2

},

{

"index": 1,

"time stamp ": "2025-03-18 02:52:07.0",

"Tx ": "Alice pays Bob 100 DSCoin",

"PrevHash":
"007edecc1e1adb97dd4da9bee199b573cfc4a53c8492607d484eb67ede6148a2",

"nonce": "288251",

"difficulty": 4

},

{

"index": 2,

"time stamp ": "2025-03-18 02:52:16.0",

"Tx ": "Bob pays Carol 20 DSCoin",

"PrevHash":
"000058570de6ee770eb05b2446bf043e370fccbde502ca0b2842c6e5da6b1c9d",

"nonce": "61854",

"difficulty": 4

},

{

"index": 3,

"time stamp ": "2025-03-18 02:52:25.0",

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

"Tx ": "Carol pays Donna 10 DSCoin",

"PrevHash": "0000746d88bc099dc538f02e83cc85f4051842558e5d53fb5f8d72e001b8790a",

"nonce": "51480",

"difficulty": 4

}

],

"chainHash": "000027946496a59dfbe4ad887acb5f29593a998a38fcd33b23c7ccd8dbede59e"

}

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

2

Verifying entire chain

Chain verification: TRUE

Total execution time required to verify the chain was 3 milliseconds

0. View basic blockchain status.

1. Add a transaction to the blockchain.

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

4

Enter block ID of block to corrupt: 2

Enter new data for block 2: Bob pays Tony 30 DSCoin

Block 2 now holds: Bob pays Tony 30 DSCoin

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

3

View the Blockchain

{

"ds_chain": [

{

"index": 0,

"time stamp ": "2025-03-18 02:50:46.0",

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

"Tx ": "Genesis",

"PrevHash": "",

"nonce": "379",

"difficulty": 2

},

{

"index": 1,

"time stamp ": "2025-03-18 02:52:07.0",

"Tx ": "Alice pays Bob 100 DSCoin",

"PrevHash":
"007edecc1e1adb97dd4da9bee199b573cfc4a53c8492607d484eb67ede6148a2",

"nonce": "288251",

"difficulty": 4

},

{

"index": 2,

"time stamp ": "2025-03-18 02:52:16.0",

"Tx ": "Bob pays Tony 30 DSCoin",

"PrevHash":
"000058570de6ee770eb05b2446bf043e370fccbde502ca0b2842c6e5da6b1c9d",

"nonce": "61854",

"difficulty": 4

},

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

{

"index": 3,

"time stamp ": "2025-03-18 02:52:25.0",

"Tx ": "Carol pays Donna 10 DSCoin",

"PrevHash": "0000746d88bc099dc538f02e83cc85f4051842558e5d53fb5f8d72e001b8790a",

"nonce": "51480",

"difficulty": 4

}

],

"chainHash": "000027946496a59dfbe4ad887acb5f29593a998a38fcd33b23c7ccd8dbede59e"

}

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

2

Verifying entire chain

Chain verification: FALSE

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

Improper hash on node 2 Does not begin with 0000

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

5

Repairing the entire chain

Total execution time required to repair the chain was 11 milliseconds

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

2

Verifying entire chain

Chain verification: TRUE

Total execution time required to verify the chain was 2 milliseconds

0. View basic blockchain status.

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

3

View the Blockchain


```
{
   "ds_chain": [
      {
         "index": 0,
         "time stamp ": "2025-03-18 02:50:46.0",
         "Tx ": "Genesis",
         "PrevHash": "",
         "nonce": "379",
         "difficulty": 2
      },
      {
         "index": 1,
         "time stamp ": "2025-03-18 02:52:07.0",
         "Tx ": "Alice pays Bob 100 DSCoin",
         "PrevHash":
```

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

"007edecc1e1adb97dd4da9bee199b573cfc4a53c8492607d484eb67ede6148a2",

"nonce": "288251",

"difficulty": 4

},

{

"index": 2,

"time stamp ": "2025-03-18 02:52:16.0",

"Tx ": "Bob pays Tony 30 DSCoin",

"PrevHash":
"000058570de6ee770eb05b2446bf043e370fccbde502ca0b2842c6e5da6b1c9d",

"nonce": "67711",

"difficulty": 4

},

{

"index": 3,

"time stamp ": "2025-03-18 02:52:25.0",

"Tx ": "Carol pays Donna 10 DSCoin",

"PrevHash":
"000038416d1aa4fd06f19ee591782e880702e77e3f7c01bb06e6034a050363e9",

"nonce": "51480",

"difficulty": 4

}

],

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

"chainHash":
"5c44bab143b01d114f612067324a8ce686225f905749308ccd5a3bcf04747e63"

}

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

6

Process finished with exit code 0

Server Output:

C:\Users\USER\.jdks\openjdk-22.0.2\bin\java.exe "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA 2024.2.0.2\lib\idea_rt.jar=53787:C:\Program Files\JetBrains\IntelliJ IDEA 2024.2.0.2\bin" -Dfile.encoding=UTF-8 -Dsun.stdout.encoding=UTF-8 -Dsun.stderr.encoding=UTF-8 -classpath C:\Users\USER\IdeaProjects\Project3Task1\target\classes;C:\Users\USER\.m2\repository\com\google\code\gson\gson\2.9.0\gson-2.9.0.jar ds.ServerTCP

Blockchain server is running

We have a visitor

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

THE JSON REQUEST MESSAGE IS SHOWN HERE

{"request":"status","difficulty":0,"blockID":0}

THE JSON RESPONSE MESSAGE IS SHOWN HERE

**{"status":"success","message":"{\"chain\":[{\"index\":0,\"timestamp\":\"Mar 18, 2025,
2:50:46 AM\",\"data\":\"Genesis\",\"previousHash\":\"\",\"nonce\":379,\"difficulty\":2}],\"chainHash\":\"007edecc1e1adb97dd4da9bee199b573cfc4a53c8492607d484eb67ede6148a2\",\"hashesPerSecond\":0}","invalidBlock":0}**

Number of Blocks on Chain == 1

We have a visitor

THE JSON REQUEST MESSAGE IS SHOWN HERE

{"request":"add transaction","data":"Alice pays Bob 100 DSCoin","difficulty":4,"blockID":0}

THE JSON RESPONSE MESSAGE IS SHOWN HERE

{"status":"success","blockChain":{"chain":[{"index":0,"timestamp":"Mar 18, 2025, 2:50:46 AM","data":"Genesis","previousHash":"","nonce":379,"difficulty":2},{"index":1,"timestamp":"Mar 18, 2025, 2:52:07 AM","data":"Alice pays Bob 100 DSCoin","previousHash":"007edecc1e1adb97dd4da9bee199b573cfc4a53c8492607d484eb67ede6148a2","nonce":288251,"difficulty":4}],"chainHash":"000058570de6ee770eb05b2446bf043e370fccbde502ca0b2842c6e5da6b1c9d","hashesPerSecond":0},"invalidBlock":0}

Number of Blocks on Chain == 2

We have a visitor

THE JSON REQUEST MESSAGE IS SHOWN HERE

{"request":"add transaction","data":"Bob pays Carol 20 DSCoin","difficulty":4,"blockID":0}

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

THE JSON RESPONSE MESSAGE IS SHOWN HERE

{"status":"success","blockChain":{"chain":[{"index":0,"timestamp":"Mar 18, 2025, 2:50:46 AM","data":"Genesis","previousHash":"","nonce":379,"difficulty":2},{"index":1,"timestamp":"Mar 18, 2025, 2:52:07 AM","data":"Alice pays Bob 100 DSCoin","previousHash":"007edecc1e1adb97dd4da9bee199b573cfc4a53c8492607d484eb67ede6148a2","nonce":288251,"difficulty":4},{"index":2,"timestamp":"Mar 18, 2025, 2:52:16 AM","data":"Bob pays Carol 20 DSCoin","previousHash":"000058570de6ee770eb05b2446bf043e370fccbde502ca0b2842c6e5da6b1c9d","nonce":61854,"difficulty":4}],"chainHash":"0000746d88bc099dc538f02e83cc85f4051842558e5d53fb5f8d72e001b8790a","hashesPerSecond":0},"invalidBlock":0}

Number of Blocks on Chain == 3

We have a visitor

THE JSON REQUEST MESSAGE IS SHOWN HERE

{"request":"add transaction","data":"Carol pays Donna 10 DSCoin","difficulty":4,"blockID":0}

THE JSON RESPONSE MESSAGE IS SHOWN HERE

{"status":"success","blockChain":{"chain":[{"index":0,"timestamp":"Mar 18, 2025, 2:50:46 AM","data":"Genesis","previousHash":"","nonce":379,"difficulty":2},{"index":1,"timestamp":"Mar 18, 2025, 2:52:07 AM","data":"Alice pays Bob 100 DSCoin","previousHash":"007edecc1e1adb97dd4da9bee199b573cfc4a53c8492607d484eb67ede6148a2","nonce":288251,"difficulty":4},{"index":2,"timestamp":"Mar 18, 2025, 2:52:16 AM","data":"Bob pays Carol 20 DSCoin","previousHash":"000058570de6ee770eb05b2446bf043e370fccbde502ca0b2842c6e5da6b1c9d","nonce":61854,"difficulty":4},{"index":3,"timestamp":"Mar 18, 2025, 2:52:25 AM","data":"Carol pays Donna 10 DSCoin","previousHash":"0000746d88bc099dc538f02e83cc85f4051842558e5d53fb5f8d72e001b8790a","nonce":51480,"difficulty":4}],"chainHash":"000027946496a59dfbe4ad887acb5f29593a998a38fcd33b23c7ccd8dbede59e","hashesPerSecond":0}

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

,"invalidBlock":0}

Number of Blocks on Chain == 4

We have a visitor

THE JSON REQUEST MESSAGE IS SHOWN HERE

{"request":"view","difficulty":0,"blockID":0}

THE JSON RESPONSE MESSAGE IS SHOWN HERE

**{"status":"success","message":"{\"chain\":[{\"index\":0,\"timestamp\":\"Mar 18, 2025,
2:50:46 AM\",\"data\":\"Genesis\",\"previousHash\":\"\",\"nonce\":379,\"difficulty\":2},{\"index\":1,\"timestamp\":\"Mar 18, 2025,
2:52:07 AM\",\"data\":\"Alice pays Bob 100
DSCoin\",\"previousHash\":\"007edecc1e1adb97dd4da9bee199b573cfc4a53c8492607d484eb67ede6148a2\",\"nonce\":288251,\"difficulty\":4},{\"index\":2,\"timestamp\":\"Mar 18, 2025, 2:52:16 AM\",\"data\":\"Bob pays Carol 20
DSCoin\",\"previousHash\":\"000058570de6ee770eb05b2446bf043e370fccbde502ca0b2842c6e5da6b1c9d\",\"nonce\":61854,\"difficulty\":4},{\"index\":3,\"timestamp\":\"Mar 18, 2025, 2:52:25 AM\",\"data\":\"Carol pays Donna 10
DSCoin\",\"previousHash\":\"0000746d88bc099dc538f02e83cc85f4051842558e5d53fb5f8d72e001b8790a\",\"nonce\":51480,\"difficulty\":4}],\"chainHash\":\"000027946496a59dfbe4ad887acb5f29593a998a38fcd33b23c7ccd8dbede59e\",\"hashesPerSecond\":0}","invalidBlock":0}**

Number of Blocks on Chain == 4

We have a visitor

THE JSON REQUEST MESSAGE IS SHOWN HERE

{"request":"verify","difficulty":0,"blockID":0}

THE JSON RESPONSE MESSAGE IS SHOWN HERE

{"status":"success","message":"TRUE","invalidBlock":0}

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

Number of Blocks on Chain == 4

We have a visitor

THE JSON REQUEST MESSAGE IS SHOWN HERE

{"request":"corrupt","data":"Bob pays Tony 30 DSCoin","difficulty":0,"blockID":2}

THE JSON RESPONSE MESSAGE IS SHOWN HERE

{"status":"success","message":"Block 2 modified.","invalidBlock":0}

Number of Blocks on Chain == 4

We have a visitor

THE JSON REQUEST MESSAGE IS SHOWN HERE

{"request":"view","difficulty":0,"blockID":0}

THE JSON RESPONSE MESSAGE IS SHOWN HERE

**{"status":"success","message":"{\"chain\":[{\"index\":0,\"timestamp\":\"Mar 18, 2025,
2:50:46 AM\",\"data\":\"Genesis\",\"previousHash\":\"\",\"nonce\":379,\"difficulty\":2},{\"index\":1,\"timestamp\":\"Mar 18, 2025,
2:52:07 AM\",\"data\":\"Alice pays Bob 100 DSCoin\",\"previousHash\":\"007edecc1e1adb97dd4da9bee199b573cfc4a53c8492607d484eb67ede6148a2\",\"nonce\":288251,\"difficulty\":4},{\"index\":2,\"timestamp\":\"Mar 18, 2025, 2:52:16 AM\",\"data\":\"Bob pays Tony 30 DSCoin\",\"previousHash\":\"000058570de6ee770eb05b2446bf043e370fccbde502ca0b2842c6e5da6b1c9d\",\"nonce\":61854,\"difficulty\":4},{\"index\":3,\"timestamp\":\"Mar 18, 2025, 2:52:25 AM\",\"data\":\"Carol pays Donna 10 DSCoin\",\"previousHash\":\"0000746d88bc099dc538f02e83cc85f4051842558e5d53fb5f8d72e001b8790a\",\"nonce\":51480,\"difficulty\":4}],\"chainHash\":\"000027946496a59dfbe4ad887acb5f29593a998a38fcd33b23c7ccd8dbede59e\",\"hashesPerSecond\":0}","invalidBlock":0}**

Number of Blocks on Chain == 4

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

We have a visitor

THE JSON REQUEST MESSAGE IS SHOWN HERE

{"request":"verify","difficulty":0,"blockID":0}

THE JSON RESPONSE MESSAGE IS SHOWN HERE

{"status":"success","message":"FALSE","invalidBlock":2}

Number of Blocks on Chain == 4

We have a visitor

THE JSON REQUEST MESSAGE IS SHOWN HERE

{"request":"hide","difficulty":0,"blockID":0}

THE JSON RESPONSE MESSAGE IS SHOWN HERE

{"status":"success","message":"Blockchain repaired.","invalidBlock":0}

Number of Blocks on Chain == 4

We have a visitor

THE JSON REQUEST MESSAGE IS SHOWN HERE

{"request":"verify","difficulty":0,"blockID":0}

THE JSON RESPONSE MESSAGE IS SHOWN HERE

{"status":"success","message":"TRUE","invalidBlock":0}

Number of Blocks on Chain == 4

We have a visitor

THE JSON REQUEST MESSAGE IS SHOWN HERE

{"request":"view","difficulty":0,"blockID":0}

THE JSON RESPONSE MESSAGE IS SHOWN HERE

{"status":"success","message":"{\"chain\":[{\"index\":0,\"timestamp\":\"Mar 18,

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

2025,

2:50:46 AM\",\"data\":\"Genesis\",\"previousHash\":\"\",\"nonce\":379,\"difficulty\":2},{\"index\":1,\"timestamp\":\"Mar 18, 2025,

2:52:07 AM\",\"data\":\"Alice pays Bob 100

DSCoin\",\"previousHash\":\"007edecc1e1adb97dd4da9bee199b573cfc4a53c8492607d484eb67ede6148a2\",\"nonce\":288251,\"difficulty\":4},{\"index\":2,\"timestamp\":\"Mar 18, 2025, 2:52:16 AM\",\"data\":\"Bob pays Tony 30

DSCoin\",\"previousHash\":\"000058570de6ee770eb05b2446bf043e370fccbde502ca0b2842c6e5da6b1c9d\",\"nonce\":67711,\"difficulty\":4},{\"index\":3,\"timestamp\":\"Mar 18, 2025, 2:52:25 AM\",\"data\":\"Carol pays Donna 10

DSCoin\",\"previousHash\":\"000038416d1aa4fd06f19ee591782e880702e77e3f7c01bb06e6034a050363e9\",\"nonce\":51480,\"difficulty\":4}],\"chainHash\":\"5c44bab143b01d114f612067324a8ce686225f905749308ccd5a3bcf04747e63\",\"hashesPerSecond\":0}","invalidBlock":0}

Number of Blocks on Chain == 4

Process finished with exit code 0

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

Task 2

Note: I do use the ChatGPT for the requirement 7:

By studying ShortMessageVerify.java and ShortMessageSign.java you will know how to compute a signature. Your solution, however, will not use the short message approach as exemplified there. Note that we are not using any Java crypto API's that abstract away the details of signing.

The thing I do is to remove the MessageDigest that is usually used for Java cryto API. I build up the function for SHA-256 make the message to be in hash, then do the SHA-256 computing.

For the requirement:

Produce a screen shot (or copy and paste) illustrating a successful execution and submit the interaction in the single pdf file as described at the end of this document. The screen shot (or copy and paste) will show three different clients interacting with the server using three distinct ID's.

I learn from ChatGPT to use the thread to execute different client's request

Project3Task2SigningClient:

```
package ds;

import com.google.gson.Gson;
import java.math.BigInteger;
import java.net.*;
import java.io.*;
import java.security.MessageDigest;
import java.util.Random;
import java.util.Scanner;

public class SigningClientTCP {
   //follow the RSA example, setting the request for signature
   private static BigInteger e, d, n;
```

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

```
    //clientID for signature
    private static String clientID;
    private static final String localHost = "localhost";
    //serverPort = 7777
    private static final int serverPort = 7777;
    //gson for geneating the request
    private static Gson gson = new Gson();

    public static void main(String[] args) {
        //get from RSAExample.java, generate the clientID
        keysGenerator();
        //print out the clientID
        keysDisplayer();
        //scanner for client input
        Scanner scanner = new Scanner(System.in);
        //user Proxy pattern to encapsulate the connection
        ClientProxy proxy = new ClientProxy(localHost, serverPort);

        System.out.println("Client is running...");

        while (true) {
            // Display the menu
            System.out.println("0. View basic blockchain status.");
            System.out.println("1. Add a transaction to the
blockchain.");
            System.out.println("2. Verify the blockchain.");
            System.out.println("3. View the blockchain.");
            System.out.println("4. Corrupt the chain.");
            System.out.println("5. Hide the corruption by repairing the
chain.");
            System.out.println("6. Exit.");
            //select the choice
            int choice = scanner.nextInt();
            scanner.nextLine();

            RequestMessage requestMessage = null;
```

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

```
            if (choice == 0) {
                //0. View basic blockchain status.
                //Set the request to status
                requestMessage = new RequestMessage("status");
            } else if (choice == 1) {
                //1. Add a transaction to the blockchain.
                //enter the difficulty and task
                System.out.print("Enter difficulty > 1: ");
                int difficulty = scanner.nextInt();
                scanner.nextLine();
                //enter the transaction
                System.out.print("Enter transaction: ");
                String data = scanner.nextLine();
                //Set the request to add transaction
                requestMessage = new RequestMessage("add transaction",
data, difficulty);
            } else if (choice == 2) {
                //2. Verify the blockchain.
                //Set the request to verify
                requestMessage = new RequestMessage("verify");
            } else if (choice == 3) {
                //3. View the blockchain.
                //Set the request to view
                requestMessage = new RequestMessage("view");
            } else if (choice == 4) {
                //4. Corrupt the chain.
                //Enter the block ID to corrupt
                System.out.print("Enter block ID of block to corrupt:
");
                int blockId = scanner.nextInt();
                scanner.nextLine();
                //Enter the new data for block
                System.out.print("Enter new data for block " + blockId
+ ": ");
                String newData = scanner.nextLine();
                //set the request to corrupt
                requestMessage = new RequestMessage("corrupt", blockId,
```

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

```
newData);
            } else if (choice == 5) {
                //5. Hide the corruption by repairing the chain.
                //Set the request to hide
                requestMessage = new RequestMessage("hide");
            } else if (choice == 6) {
                //6. Exit.
                proxy.close();
                return;
            } else {
                System.out.println("Invalid choice. Try again.");
                continue;
            }
            //Set the information details in requestMessage
            requestMessage.setClientID(clientID);
            requestMessage.setPublicKey(e, n);

requestMessage.setSignature(messageSigning(requestMessage.getConcaten
atedValues()));
            //make it to json format
            String jsonRequest = gson.toJson(requestMessage);

            System.out.println("Sent: " + jsonRequest);
            //use proxy to get the Response from server
            String responseData = proxy.sendRequest(requestMessage);
            if (responseData != null) {
                ResponseMessage responseMessage =
gson.fromJson(responseData, ResponseMessage.class);
                //Print out the message from server
                System.out.println("Response: ");
                System.out.println(responseMessage.getMessage());
            } else {
                System.out.println("Error: No response received from
the server.");
            }
        }
    }
```

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

```
    //From RSAExample.java. used for generate the client ID
    private static void keysGenerator() {
        Random rnd = new Random();
        BigInteger p = new BigInteger(400, 100, rnd);
        BigInteger q = new BigInteger(400, 100, rnd);
        n = p.multiply(q);
        BigInteger phi =
(p.subtract(BigInteger.ONE)).multiply(q.subtract(BigInteger.ONE));
        e = new BigInteger("65537");
        d = e.modInverse(phi);
        clientID = getClientId();
    }
    //display the key
    private static void keysDisplayer() {
        System.out.println("Client Public Key: e = " + e + ", n = " +
n);
        System.out.println("Client Private Key: d = " + d + ", n = " +
n);
        System.out.println("Client ID: " + clientID);
    }
    /*
    * The client's ID will be formed by taking the least significant
20 bytes of the hash of the client's public key.
    *  Note: an RSA public key is the pair e and n. Prior to hashing,
you will combine these two integers with concatenation.
    *  The ID is computed in the client code. As in Bitcoin or
Ethereum, the user's ID is derived from the public key.
    * */
    private static String getClientId() {
        try {
            //SHA-256 computing
            MessageDigest md = MessageDigest.getInstance("SHA-256");
            //use e and n to do the update
            md.update((e.toString() + n.toString()).getBytes());
            //get the byte of the md
            byte[] hash = md.digest();
            //Generate the client ID by taking the last 20 bytes of the
```

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

```
SHA-256 hash of the public key
            //BigInteger(1, hash): to ensure the byte array hash is
always positive
            //toString(16): hexadecimal
            return new BigInteger(1,
hash).toString(16).substring(hash.length - 20);
        } catch (Exception ex) {
            throw new RuntimeException(ex);
        }
    }

    /*
    * By studying ShortMessageVerify.java and ShortMessageSign.java
you will know how to compute a signature.
    * Your solution, however, will not use the short message approach
as exemplified there.
    * Note that we are not using any Java crypto API's that abstract
away the details of signing.
    * */
    private static String messageSigning(String message) {
        try {
            // Step 1: Compute SHA-256 hash of the message (without
Java Crypto APIs)
            byte[] messageHash = sha256Hash(message.getBytes("UTF-8"));

            // Step 2: Convert hash to a BigInteger (ensuring it's
positive)
            BigInteger hashInt = new BigInteger(1, messageHash);

            // Step 3: Encrypt the hash using the private key (RSA
signing)
            BigInteger signedValue = hashInt.modPow(d, n);

            // Step 4: Return the signature as a base-10 string
            return signedValue.toString();
        } catch (Exception ex) {
            throw new RuntimeException(ex);
```

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

```
        }
    }
    // Manual SHA-256 implementation
    //  SHA-256 Hash implementation to replaces Java's MessageDigest
    private static byte[] sha256Hash(byte[] data) {
        int hash = 0;
        for (byte b : data) {
            // Bitwise mixing for hash uniqueness
            hash = (hash * 31) ^ (b & 0xFF);
        }
        byte[] hashBytes = BigInteger.valueOf(hash).toByteArray();

        // Ensure a fixed 32-byte output like SHA-256
        byte[] finalHash = new byte[32];
        System.arraycopy(hashBytes, 0, finalHash, 32 -
hashBytes.length, hashBytes.length);
        return finalHash;
    }

}
//Handling the request
class RequestMessage {
    //variables for requestMessage
    private String request;
    private String data;
    private int difficulty;
    private int blockID;
    private String clientID;
    private String signature;
    private BigInteger e;
    private BigInteger n;

    // Constructors
    //For selection without setting values
    public RequestMessage(String request) {
        this.request = request;
    }
```

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

```
    //For selection need to add the block to block chain
    public RequestMessage(String request, String data, int
difficulty) {
        this.request = request;
        this.data = data;
        this.difficulty = difficulty;
    }
    //For selection need to corrupt the block
    public RequestMessage(String request, int blockID, String data) {
        this.request = request;
        this.blockID = blockID;
        this.data = data;
    }

    // Getters and setters
    public String getRequest() { return request; }
    public String getData() { return data; }
    public int getDifficulty() { return difficulty; }
    public int getBlockID() { return blockID; }
    public String getSignature() { return signature; }
    public BigInteger getPublicKeyE() { return e; }
    public BigInteger getPublicKeyN() { return n; }
    public String getClientID(){
        return this.clientID;
    }

    public void setClientID(String clientID) { this.clientID =
clientID; }
    public void setSignature(String signature) { this.signature =
signature; }
    public void setPublicKey(BigInteger e, BigInteger n) {
        this.e = e;
        this.n = n;
    }
    //input the message such like request, data, clientID...etc to
create the signature
    public String getConcatenatedValues() {
```

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

```
        StringBuilder sb = new StringBuilder();
        sb.append(request);
        if (data != null) sb.append(data);
        if (difficulty > 0) sb.append(difficulty);
        if (blockID >= 0) sb.append(blockID);
        sb.append(clientID);
        return sb.toString();
    }
}
//Handle server's response
class ResponseMessage {
    private String status;
    private String message;

    // Constructor
    public ResponseMessage(String status, String message) {
        this.status = status;
        this.message = message;
    }

    // Getters and Setters
    public String getStatus() { return status; }
    public String getMessage() { return message; }
}
//Handle the client's connection, communication to server
class ClientProxy {
    private Socket socket;
    private PrintWriter out;
    private BufferedReader in;

    private Gson gson = new Gson();
    //Constructor to connect to server
    public ClientProxy(String serverAddress, int serverPort) {
        try {
            this.socket = new Socket(serverAddress, serverPort);
            this.out = new PrintWriter(socket.getOutputStream(), true);
            this.in = new BufferedReader(new
```

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

```
InputStreamReader(socket.getInputStream()));
        } catch (IOException e) {
            e.printStackTrace();
        }
    }
    //Send out the request and receive the response from server
    public String sendRequest(RequestMessage request) {
        try {
            String jsonRequest = gson.toJson(request);
            out.println(jsonRequest);
            out.flush();
            return in.readLine();
        } catch (IOException e) {
            e.printStackTrace();
            return null;
        }
    }
    //close the connection once client select Exit
    public void close() {
        try {
            socket.close();
            out.close();
            in.close();
        } catch (IOException e) {
            e.printStackTrace();
        }
    }
}
```

Project3Task2VerifyingServer

```
package ds;

import com.google.gson.Gson;
import java.math.BigInteger;
import java.net.*;
```

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

```
import java.io.*;

public class VerifyingServerTCP {
    // Initialize the blockchain
    private static BlockChain blockChain = new BlockChain();

    public static void main(String[] args) {
        //set the port to 7777
        try (ServerSocket serverSocket = new ServerSocket(7777)) {
            System.out.println("Blockchain server is running...");
            while (true) {
                // Accept client connections and handle them in a new
thread
                Socket clientSocket = serverSocket.accept();
                System.out.println("New client connected...");

                // Create a new thread for each client
                new Thread(new ServerProxy(clientSocket,
blockChain)).start();
            }
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}

// Proxy pattern for server & client communication. Runnable class to
handle each client independently
class ServerProxy implements Runnable {
    //Necessary variable for ClientHandler
    private Socket clientSocket;
    private BlockChain blockChain;
    private PrintWriter out;
    private BufferedReader in;
    private Gson gson = new Gson();

    //Connect to client
```

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

```
    public ServerProxy(Socket socket, BlockChain blockChain) {
        this.clientSocket = socket;
        this.blockChain = blockChain;
        try {
            this.out = new PrintWriter(clientSocket.getOutputStream(),
true);
            this.in = new BufferedReader(new
InputStreamReader(clientSocket.getInputStream()));
        } catch (IOException e) {
            System.out.println("Error initializing client handler: " +
e.getMessage());
        }
    }
    //run() to execute multiple client's request
    @Override
    public void run() {
        try {
            //Get the start time
            long startTime = System.currentTimeMillis();
            while (true) {
                //Get the request
                String jsonRequest = in.readLine();
                //if there is no request, break
                if (jsonRequest == null) break;
                //Build up the json request
                RequestMessage request = gson.fromJson(jsonRequest,
RequestMessage.class);
                System.out.println("We have a visitor");
                System.out.println("THE JSON REQUEST MESSAGE IS SHOWN
HERE:");
                System.out.println(gson.toJson(request));

                // Verify the signature before processing the request
                if (!verifySignature(request)) {
                    ResponseMessage errorResponse = new
ResponseMessage("failed", "Signature verification failed!");
                    out.println(gson.toJson(errorResponse));
```

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

```
                continue;
            }

            // Process the request and generate a response
            ResponseMessage response =
clientRequestHandler(request, startTime);
            String jsonResponse = gson.toJson(response);
            //Print out response from server
            System.out.println("THE JSON RESPONSE MESSAGE IS SHOWN
HERE:");
            System.out.println(jsonResponse);
            //Send back to client
            out.println(jsonResponse);
            System.out.println("Number of Blocks on Chain == " +
blockChain.getChainSize());
          }
       } catch (IOException e) {
          System.out.println("Client connection lost.");
       } finally {
          closeConnection();
       }
    }
    //Process client's request, startTime for calculating the total
time
    private ResponseMessage clientRequestHandler(RequestMessage
request, long startTime) {
       //StringBuilder for create message
       StringBuilder sb = new StringBuilder();
       ResponseMessage response;

       if (request.getRequest().equals("status")) {
          //Put the details of the blockChain into response, just
like task0 and task1
          sb.append("Current size of chain:
").append(blockChain.getChainSize()).append("\n");
          sb.append("Difficulty of most recent block:
").append(blockChain.getLatestBlock().getDifficulty()).append("\n");
```

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

```
            sb.append("Total difficulty for all blocks:
").append(blockChain.getTotalDifficulty()).append("\n");
            sb.append("Approximate hashes per second:
").append(blockChain.getHashesPerSecond()).append("\n");
            sb.append("Expected total hashes required:
").append(blockChain.getTotalExpectedHashes()).append("\n");
            sb.append("Nonce for most recent block:
").append(blockChain.getLatestBlock().getNonce()).append("\n");
            sb.append("Chain hash:
").append(blockChain.getChainHash()).append("\n");
            response = new ResponseMessage("success", sb.toString());

        } else if (request.getRequest().equals("add transaction")) {
            //add the block to blockChain
            blockChain.addBlock(new Block(blockChain.getChainSize(),
blockChain.getTime(), request.getData(), request.getDifficulty()));
            //get the execution Time
            long executionTime = System.currentTimeMillis() -
startTime;
            sb.append("Total execution time to add this block was
").append(executionTime).append(" milliseconds\n");
            response = new ResponseMessage("success", sb.toString());

        } else if (request.getRequest().equals("verify")) {
            //Check if the Chain is valid or not
            //if true: get the verification time, and send to response
            //if false: point out the node that is invalid
            if (blockChain.isChainValid().equals("TRUE")) {
                long verificationTime = System.currentTimeMillis() -
startTime;
                sb.append("Verifying entire chain\n");
                sb.append("Chain verification: TRUE\n");
                sb.append("Total execution time required to verify the
chain was ").append(verificationTime).append(" milliseconds\n");
                response = new ResponseMessage("success",
sb.toString());
            } else {
```

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

```
                sb.append("Chain verification: FALSE\n");
                sb.append("Improper hash on node
").append(blockChain.inValidBlock()).append(" Does not begin with
0000\n");
                response = new ResponseMessage("failed",
sb.toString());
            }

        } else if (request.getRequest().equals("view")) {
            //get all details of blockChain and put into message
            sb.append("View the Blockchain\n");
            sb.append(blockChain.toString()).append("\n");
            response = new ResponseMessage("success", sb.toString());

        } else if (request.getRequest().equals("corrupt")) {
            //corrupt the block that request by client
            sb.append("Block ").append(request.getBlockID()).append("
now holds: ").append(request.getData()).append("\n");

blockChain.getBlock(request.getBlockID()).setData(request.getData());
            response = new ResponseMessage("success", sb.toString());

        } else if (request.getRequest().equals("hide")) {
            //repair the chain and show out the repairing time
            blockChain.repairChain();
            long repairTime = System.currentTimeMillis() - startTime;
            sb.append("Repairing the entire chain\n");
            sb.append("Total execution time required to repair the
chain was ").append(repairTime).append(" milliseconds\n");
            response = new ResponseMessage("success", sb.toString());

        } else {
            response = new ResponseMessage("failed", "Invalid
request!");
        }
        return response;
    }
```

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

```
    // Learn from ShortMessageVerify
    /*
     * By studying ShortMessageVerify.java and ShortMessageSign.java
you will know how to compute a signature.
     * Your solution, however, will not use the short message approach
as exemplified there.
     * Note that we are not using any Java crypto API's that abstract
away the details of signing.
     * */
    private boolean verifySignature(RequestMessage request) {
        try {
            // Step 1: Compute SHA-256 hash of the original message
            byte[] messageHash =
sha256Hash(request.getConcatenatedValues().getBytes("UTF-8"));
            // Step 2: Convert the computed hash into a BigInteger
(ensuring it's positive)
            BigInteger hashInt = new BigInteger(1, messageHash);
            // Step 3: Retrieve the signature and decrypt it using the
public key
            BigInteger decryptedHash = new
BigInteger(request.getSignature())
                    .modPow(request.getPublicKeyE(),
request.getPublicKeyN());

            System.out.println("Computed hash: " + hashInt);
            System.out.println("Decrypted signature: " +
decryptedHash);

            // Step 4: Compare the decrypted hash with the computed
hash
            return hashInt.equals(decryptedHash);
        } catch (Exception ex) {
            return false;
        }
    }
    //Manual SHA-256 implementation
    //  SHA-256 Hash implementation to replaces Java's MessageDigest
```

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

```
    private static byte[] sha256Hash(byte[] data) {
        int hash = 0;
        for (byte b : data) {
            // Bitwise mixing for hash uniqueness
            hash = (hash * 31) ^ (b & 0xFF);
        }
        byte[] hashBytes = BigInteger.valueOf(hash).toByteArray();
        // Ensure a fixed 32-byte output like SHA-256
        byte[] finalHash = new byte[32];
        System.arraycopy(hashBytes, 0, finalHash, 32 -
hashBytes.length, hashBytes.length);
        return finalHash;
    }
    //close the server
    private void closeConnection() {
        try {
            if (clientSocket != null) clientSocket.close();
            if (out != null) out.close();
            if (in != null) in.close();
        } catch (IOException e) {
            System.out.println("Error closing connection: " +
e.getMessage());
        }
    }
}
```

SigningClientTCP's Output:

Client 1:

C:\Users\USER\.jdks\openjdk-22.0.2\bin\java.exe "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA 2024.2.0.2\lib\idea_rt.jar=53737:C:\Program Files\JetBrains\IntelliJ IDEA 2024.2.0.2\bin" -Dfile.encoding=UTF-8 -Dsun.stdout.encoding=UTF-8 -Dsun.stderr.encoding=UTF-8 -classpath C:\Users\USER\IdeaProjects\Project3Task2\target\classes;C:\Users\USER\.m2\repository\com\google\code\gson\gson\2.9.0\gson-2.9.0.jar ds.SigningClientTCP

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

Client Public Key: e = 65537, n = 458059865128652378038727029632542768974397082279984574317839934053260764280652910602838420946732019319243824177647565974244041819103379596784817871676035427303109317495453804009227304064078141252454872001223878025491814304179914697748331195 3

Client Private Key: d = 135278524948732269968713271259561244693828761272702464804946700697329174247402796660175742518333128369073412224520021353177532991549240133013410812559839107511075579680548694624317454204774436346089111420736026297560208266289571878094497339 3, n = 458059865128652378038727029632542768974397082279984574317839934053260764280652910602838420946732019319243824177647565974244041819103379596784817871676035427303109317495453804009227304064078141252454872001223878025491814304179914697748331195 3

Client ID: 03ee0037c93a333ef371d410cbcb4138d98134772102d51aa41

Client is running...

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

0

Sent:
{"request":"status","difficulty":0,"blockID":0,"clientID":"03ee0037c93a333ef371d410cbcb4138d98134772102d51aa41","signature":"434502712166497811938292751

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

6930910392652050679827379398631945211207263333142688051628045779303258538048320131279040511584269936807028500267888076906488115070163387858072670575004933836159970581437416378768526839430347073903164578892901653293","e":65537,"n":458059865128652378038727029632542768974397082279984574317839934053260764280652910602838420946732019319243824177647565974244041819103379596784817871676035427303109317495453804009227304064078141252454872001223878025491814304179914697748331195 3}

Response:

Current size of chain: 1

Difficulty of most recent block: 2

Total difficulty for all blocks: 2

Approximate hashes per second: 0

Expected total hashes required: 256.0

Nonce for most recent block: 246

Chain hash:
00de2df1f886a84cb09e7f118e9772a19b29ac2e1666bdf132f2d86909b1e4a7

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

1

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

Enter difficulty > 1: 4

Enter transaction: Alice pays Bob 100 DSCoin

Sent: {"request":"add transaction","data":"Alice pays Bob 100 DSCoin","difficulty":4,"blockID":0,"clientID":"03ee0037c93a333ef371d410cbcb4138d98134772102d51aa41","signature":"30308899969078645593683828455033347300331557370933805428181691522254116022442802148259521036074330740525620373028647436931705294591713448082534958832397796761462098169014770037680267973880937292903930343212273273457124453906169935243657146361","e":65537,"n":458059865128652378038727029632542768974397082279984574317839934053260764280652910602838420946732019319243824177647565974244041819103379596784817871676035427303109317495453804009227304064078141252454872001223878025491814304179914697748331195 3}

Response:

Total execution time to add this block was 18343 milliseconds

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

1

Enter difficulty > 1: 4

Enter transaction: Bob pays Carol 20 DSCoin

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

Sent: {"request":"add transaction","data":"Bob pays Carol 20 DSCoin","difficulty":4,"blockID":0,"clientID":"03ee0037c93a333ef371d410cbcb4138d98134772102d51aa41","signature":"35340744438234762266906470514929624062951959346981988561394028734172440635378848009912195717683832240059820253948492330430591036148583967837491448947059556796974182418572721066573621219045144354201806279210839683864587635623241700336484837666","e":65537,"n":458059865128652378038727029632542768974397082279984574317839934053260764280652910602838420946732019319243824177647565974244041819103379596784817871676035427303109317495453804009227304064078141252454872001223878025491814304179914697748331195 3}

Response:

Total execution time to add this block was 26838 milliseconds

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

1

Enter difficulty > 1: 4

Enter transaction: Carol pays Donna 10 DSCoin

Sent: {"request":"add transaction","data":"Carol pays Donna 10 DSCoin","difficulty":4,"blockID":0,"clientID":"03ee0037c93a333ef371d410cbcb4138d98134772102d51aa41","signature":"394850918089711359433536025545887008

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

7116605462716157222584147236542144412287866913287324226540470045822150356981759685941265737586152303045277949433617601713211463303373468529130325764455967191221133005591370391405905355224879863423786766119","e":65537,"n":458059865128652378038727029632542768974397082274317839934053260764280652910602838420946732019319243824177647565974244041819103379596784817871676035427303109317495453804009227304064078141252454872001223878025491814304179914697748331195 3}

Response:

Total execution time to add this block was 34499 milliseconds

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

3

Sent:

{"request":"view","difficulty":0,"blockID":0,"clientID":"03ee0037c93a333ef371d410cbcb4138d98134772102d51aa41","signature":"289460066051856012076743226715439865707189595391895296554761456262462944178721310076029635258998926069431168474301007582267214613989148325782406218018874907725207163248037245808047072166544302910458441772259951209506291694347236585091236672 1","e":65537,"n":458059865128652378038727029632542768974397082279984574317839934053260764280652910602838420946732019319243824177647565974244041819103379596784817871676035427303109317495453804009227304064078141252454872001223878025491814304179914697748331195 3}

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

040640781412524548720012238780254918143041799146977483311953}

Response:

View the Blockchain


```
{
   "ds_chain": [
      {
         "index": 0,
         "time stamp ": "2025-03-18 02:39:57.743",
         "Tx ": "Genesis",
         "PrevHash": "",
         "nonce": "246",
         "difficulty": 2
      },
      {
         "index": 1,
         "time stamp ": "2025-03-18 02:40:24.637",
         "Tx ": "Alice pays Bob 100 DSCoin",
         "PrevHash":
"00de2df1f886a84cb09e7f118e9772a19b29ac2e1666bdf132f2d86909b1e4a7",
         "nonce": "69462",
         "difficulty": 4
      },
      {
```

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

"index": 2,

"time stamp ": "2025-03-18 02:40:33.492",

"Tx ": "Bob pays Carol 20 DSCoin",

"PrevHash": "0000a6363742fe2ec6dde1e9c5adbbcbfc04c4fbffe15dbbedc83d93cfd6f7c7",

"nonce": "3899",

"difficulty": 4

},

{

"index": 3,

"time stamp ": "2025-03-18 02:40:41.072",

"Tx ": "Carol pays Donna 10 DSCoin",

"PrevHash": "000056dc8a4ba49c9133527f5fdfd7ebf8a11b1bc48562aaa932a514ef60bce5",

"nonce": "36221",

"difficulty": 4

}

],

"chainHash": "0000c6e10ef23ebd325563f24e69ae0b17ee3759773a6119c06d308446834f1a"

}

0. View basic blockchain status.

1. Add a transaction to the blockchain.

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

2

Sent:

{"request":"verify","difficulty":0,"blockID":0,"clientID":"03ee0037c93a333ef371d410cbcb4138d98134772102d51aa41","signature":"167109993880994211425866479951178869939504141331523256049914159561871666277050655199098678610667398152333648419798520165860729676498803067744162523183753767956053632847209822103148044721083820720934799089406806699537596400791486163541134892 2","e":65537,"n":458059865128652378038727029632542768974397082279984574317839934053260764280652910602838420946732019319243824177647565974244041819103379596784817871676035427303109317495453804009227304064078141252454872001223878025491814304179914697748331195 3}

Response:

Verifying entire chain

Chain verification: TRUE

Total execution time required to verify the chain was 39775 milliseconds

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

4

Enter block ID of block to corrupt: 2

Enter new data for block 2: Bob pays Tony 30 DSCoin

Sent: {"request":"corrupt","data":"Bob pays Tony 30 DSCoin","difficulty":0,"blockID":2,"clientID":"03ee0037c93a333ef371d410cbcb4138d98134772102d51aa41","signature":"829673892454249220548004090798303449767179547681425742964034493768745066458147141254641202426423887155426069480081437703500058780985177130829029240585948806166521022445820345018275069176151395488212277280306732479141853702232588205834945822","e":65537,"n":458059865128652378038727029632542768974397082279984574317839934053260764280652910602838420946732019319243824177647565974244041819103379596784817871676035427303109317495453804009227304064078141252454872001223878025491814304179914697748331195 3}

Response:

Block 2 now holds: Bob pays Tony 30 DSCoin

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

6. Exit.

3

Sent:

{"request":"view","difficulty":0,"blockID":0,"clientID":"03ee0037c93a333ef371d410cbcb4138d98134772102d51aa41","signature":"2894600660518560120767432267154398657071895953918952965547614562624629441787213100760296352589989260694311684743010075822672146139891483257824062180188749077252071632480372458080470721665443029104584417722599512095062916943472365850912366721","e":65537,"n":458059865128652378038727029632542768974397082279984574317839934053260764280652910602838420946732019319243824177647565974244041819103379596784817871676035427303109317495453804009227304064078141252454872001223878025491814304179914697748331195 3}

Response:

View the Blockchain

{

"ds_chain": [

{

"index": 0,

"time stamp ": "2025-03-18 02:39:57.743",

"Tx ": "Genesis",

"PrevHash": "",

"nonce": "246",

"difficulty": 2

},

{

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

"index": 1,

"time stamp ": "2025-03-18 02:40:24.637",

"Tx ": "Alice pays Bob 100 DSCoin",

"PrevHash": "00de2df1f886a84cb09e7f118e9772a19b29ac2e1666bdf132f2d86909b1e4a7",

"nonce": "69462",

"difficulty": 4

},

{

"index": 2,

"time stamp ": "2025-03-18 02:40:33.492",

"Tx ": "Bob pays Tony 30 DSCoin",

"PrevHash": "0000a6363742fe2ec6dde1e9c5adbbcbfc04c4fbffe15dbbedc83d93cfd6f7c7",

"nonce": "3899",

"difficulty": 4

},

{

"index": 3,

"time stamp ": "2025-03-18 02:40:41.072",

"Tx ": "Carol pays Donna 10 DSCoin",

"PrevHash": "000056dc8a4ba49c9133527f5fdfd7ebf8a11b1bc48562aaa932a514ef60bce5",

"nonce": "36221",

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

"difficulty": 4

}

],

"chainHash":
"0000c6e10ef23ebd325563f24e69ae0b17ee3759773a6119c06d308446834f1a"

}

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

2

Sent:

{"request":"verify","difficulty":0,"blockID":0,"clientID":"03ee0037c93a333ef371d4
10cbcb4138d98134772102d51aa41","signature":"16710999388099421142586647 9
95117886993950414133152325604991415956187166627705065519909867861066
73981523336484197985201658607296764988030677441625231837537679560536
32847209822103148044721083820720934799089406806699537596400791486163
5411348922","e":65537,"n":458059865128652378038727029632542768974397708
22799845743178399340532607642806529106028384209467320193192438241776
47565974244041819103379596784817871676035427303109317495453804009227
30406407814125245487200122387802549181430417991469774833119 53}

Response:

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

Chain verification: FALSE

Improper hash on node 2 Does not begin with 0000

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

5

Sent:

{"request":"hide","difficulty":0,"blockID":0,"clientID":"03ee0037c93a333ef371d410cbcb4138d98134772102d51aa41","signature":"279672779487993891534673303541874803866841725641062650198373877387582895750811710140260322327884447251397332560097696717175003982068035713121151007468024749272503884736460871520160139564528808028128084794365599421179939944306790382468341400 9","e":65537,"n":458059865128652378038727029632542768974397082279984574317839934053260764280652910602838420946732019319243824177647565974244041819103379596784817871676035427303109317495453804009227304064078141252454872001223878025491814304179914697748331195 3}

Response:

Repairing the entire chain

Total execution time required to repair the chain was 53452 milliseconds

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

2

Sent:

{"request":"verify","difficulty":0,"blockID":0,"clientID":"03ee0037c93a333ef371d4
10cbcb4138d98134772102d51aa41","signature":"16710999388099421142586647 9
95117886993950414133152325604991415956187166627705065519909867861066
73981523336484197985201658607296764988030677441625231837537679560536
32847209822103148044721083820720934799089406806699537596400791486163
5411348922","e":65537,"n":458059865128652378038727029632542768974397 08
22799845743178399340532607642806529106028384209467320193192438241776
47565974244041819103379596784817871676035427303109317495453804009227
30406407814125245487200122387802549181430417991469774833119 53}

Response:

Verifying entire chain

Chain verification: TRUE

Total execution time required to verify the chain was 58447 milliseconds

0. View basic blockchain status.

1. Add a transaction to the blockchain.

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

3

Sent:

{"request":"view","difficulty":0,"blockID":0,"clientID":"03ee0037c93a333ef371d41
0cbcb4138d98134772102d51aa41","signature":"28946006605185601207674322671
54398657071895953918952965547614562624629441787213100760296352589989
26069431168474301007582267214613989148325782406218018874907725207163
24803724580804707216654430291045844177225995120950629169434723658509
12366721","e":65537,"n":458059865128652378038727029632542768974397082
27998457431783993405326076428065291060283842094673201931924382417764
75659742440418191033795967848178716760354273031093174954538040092273
04064078141252454872001223878025491814304179914697748331195 3}

Response:

View the Blockchain

{

"ds_chain": [

{

"index": 0,

"time stamp ": "2025-03-18 02:39:57.743",

"Tx ": "Genesis",

"PrevHash": "",

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

"nonce": "246",

"difficulty": 2

},

{

"index": 1,

"time stamp ": "2025-03-18 02:40:24.637",

"Tx ": "Alice pays Bob 100 DSCoin",

"PrevHash":
"00de2df1f886a84cb09e7f118e9772a19b29ac2e1666bdf132f2d86909b1e4a7",

"nonce": "69462",

"difficulty": 4

},

{

"index": 2,

"time stamp ": "2025-03-18 02:40:33.492",

"Tx ": "Bob pays Tony 30 DSCoin",

"PrevHash":
"0000a6363742fe2ec6dde1e9c5adbbcbfc04c4fbffe15dbbedc83d93cfd6f7c7",

"nonce": "50929",

"difficulty": 4

},

{

"index": 3,

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

"time stamp ": "2025-03-18 02:40:41.072",

"Tx ": "Carol pays Donna 10 DSCoin",

"PrevHash": "000078b73644bccd2b8eabad72397900e6465628f1b3f7eeda0ac97d720e597c",

"nonce": "36221",

"difficulty": 4

}

],

"chainHash": "f54354c322c1649a098978d8b9a63530ad6a1bb58adc74fe32403a0a5e76a127"

}

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

6

Process finished with exit code 0

Client 2:

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

C:\Users\USER\.jdks\openjdk-22.0.2\bin\java.exe "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA 2024.2.0.2\lib\idea_rt.jar=53733:C:\Program Files\JetBrains\IntelliJ IDEA 2024.2.0.2\bin" -Dfile.encoding=UTF-8 -Dsun.stdout.encoding=UTF-8 -Dsun.stderr.encoding=UTF-8 -classpath C:\Users\USER\IdeaProjects\Project3Task2\target\classes;C:\Users\USER\.m2\repository\com\google\code\gson\gson\2.9.0\gson-2.9.0.jar ds.SigningClientTCP2

Client Public Key: e = 65537, n = 448874687648573780753681236128106014244145624770942349574946099856524338242562799230182230664951690337026670615421769641635346658397432304304002590759683036458351592105322698809372417268313551919408060699332727705174541898691644954107401676 1

Client Private Key: d = 210804051702821365397207700772858795892493004550102440380513161441385874886393912365633896812577370419656176941291960663704195770366756264484864779788732814039145183263336485770178380064582076145869860654362128517343934284793998026185960576 1, n = 448874687648573780753681236128106014244145624770942349574946099856524338242562799230182230664951690337026670615421769641635346658397432304304002590759683036458351592105322698809372417268313551919408060699332727705174541898691644954107401676 1

Client ID: ba77d68adec2806639c519a16341e7621b0dac357be3e11169ef

Client is running...

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

6. Exit.

0

Sent:

{"request":"status","difficulty":0,"blockID":0,"clientID":"ba77d68adec2806639c51
9a16341e7621b0dac357be3e11169ef","signature":"16524460757601956785055 62
96901625915777689720789251298208998267254273867896381205743465815354
80391898604164362967205486072384054963798109054045334580055101021406
69352175495294502293497320964549328969809969221192215760565386561883
819423025419","e":65537,"n":448874687648573780753681236128106014244145
62477094234957494609985652433824256279923018223066495169033702667061
54217696416353466583974323043040025907596830364583515921053226988093
72417268313551919408060699332727705174541898691644954107401676 1}

Response:

Current size of chain: 4

Difficulty of most recent block: 4

Total difficulty for all blocks: 14

Approximate hashes per second: 0

Expected total hashes required: 196864.0

Nonce for most recent block: 36221

Chain hash:
f54354c322c1649a098978d8b9a63530ad6a1bb58adc74fe32403a0a5e76a127

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

1

Enter difficulty > 1: 4

Enter transaction: Alice pays Bob 100 DSCoin

Sent: {"request":"add transaction","data":"Alice pays Bob 100 DSCoin","difficulty":4,"blockID":0,"clientID":"ba77d68adec2806639c519a16341e7621b0dac357be3e11169ef","signature":"32276985700204668786784623904663968637389519233762492525748755734490591455838488220113432733750595024102577907834209368708319975707769950024973362400023708195781642771875971868047178382002732314363850221032872137557155362114253403173542502 94","e":65537,"n":44887468764857378075368123612810601424414562477094234957494609985652433824256279923018223066495169033702667061542176964163534665839743230430400259075968303645835159210532269880937241726831355191940806069933272770517454189869164495410740167 61}

Response:

Total execution time to add this block was 100830 milliseconds

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

6. Exit.

1

Enter difficulty > 1: 4

Enter transaction: Bob pays Carol 20 DSCoin

Sent: {"request":"add transaction","data":"Bob pays Carol 20 DSCoin","difficulty":4,"blockID":0,"clientID":"ba77d68adec2806639c519a16341e7621b0dac357be3e11169ef","signature":"18614193892217844696529206635708740490290660067773728740359871137149010021745029797234008063206458038230485532481557481845355779534253315086613519885786197650253851789302519620162063936706481729689040974288593097917383826669500135833989278 6","e":65537,"n":448874687648573780753681236128106014244145624770942349574946099856524338242562799230182230664951690337026670615421769641635346658397432304304002590759683036458351592105322698809372417268313551919408060699332727705174541898691644954107401676 1}

Response:

Total execution time to add this block was 108511 milliseconds

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

1

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

Enter difficulty > 1: 4

Enter transaction: Carol pays Donna 10 DSCoin

Sent: {"request":"add transaction","data":"Carol pays Donna 10 DSCoin","difficulty":4,"blockID":0,"clientID":"ba77d68adec2806639c519a16341e7621b0dac357be3e11169ef","signature":"619509344649138424533196414170813917013681767566175956931287742436286080521849220846959402096970154049097393012924514204870029943686489901712379236291833122779372324443498542042859942545313249202020633467639129196874354437255821661625123567","e":65537,"n":448874687648573780753681236128106014244145624770942349574946099856524338242562799230182230664951690337026670615421769641635346658397432304304002590759683036458351592105322698809372417268313551919408060699332727705174541898691644954107401676 1}

Response:

Total execution time to add this block was 116002 milliseconds

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

3

Sent: {"request":"view","difficulty":0,"blockID":0,"clientID":"ba77d68adec2806639c519a16341e7621b0dac357be3e11169ef","signature":"100612469282391272199364 06

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

00570763608412118324341812768795256072723252041900455044603331836897295173002167177455323868628633252721881712324624150971623729932276216105734048145216237655291276912138793944247527444520930983361446847877199572404","e":65537,"n":448874687648573780753681236128106014244145624770942349574946099856524338242562799230182230664951690337026670615421769641635346658397432304304002590759683036458351592105322698809372417268313551919408060699332727705174541898691644954107401676 1}

Response:

View the Blockchain


```
{
   "ds_chain": [
      {
         "index": 0,
         "time stamp ": "2025-03-18 02:39:57.743",
         "Tx ": "Genesis",
         "PrevHash": "",
         "nonce": "246",
         "difficulty": 2
      },
      {
         "index": 1,
         "time stamp ": "2025-03-18 02:40:24.637",
         "Tx ": "Alice pays Bob 100 DSCoin",
         "PrevHash":
"00de2df1f886a84cb09e7f118e9772a19b29ac2e1666bdf132f2d86909b1e4a7",
```

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

"nonce": "69462",

"difficulty": 4

},

{

"index": 2,

"time stamp ": "2025-03-18 02:40:33.492",

"Tx ": "Bob pays Tony 30 DSCoin",

"PrevHash":
"0000a6363742fe2ec6dde1e9c5adbbcbfc04c4fbffe15dbbedc83d93cfd6f7c7",

"nonce": "50929",

"difficulty": 4

},

{

"index": 3,

"time stamp ": "2025-03-18 02:40:41.072",

"Tx ": "Carol pays Donna 10 DSCoin",

"PrevHash":
"000078b73644bccd2b8eabad72397900e6465628f1b3f7eeda0ac97d720e597c",

"nonce": "36221",

"difficulty": 4

},

{

"index": 4,

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

"time stamp ": "2025-03-18 02:41:45.381",

"Tx ": "Alice pays Bob 100 DSCoin",

"PrevHash": "f54354c322c1649a098978d8b9a63530ad6a1bb58adc74fe32403a0a5e76a127",

"nonce": "108057",

"difficulty": 4

},

{

"index": 5,

"time stamp ": "2025-03-18 02:41:53.142",

"Tx ": "Bob pays Carol 20 DSCoin",

"PrevHash": "0000ed0c56e5a5addefc89f54d83bbc0151faf5884e43480e99f3d6369f561f0",

"nonce": "31620",

"difficulty": 4

},

{

"index": 6,

"time stamp ": "2025-03-18 02:42:00.613",

"Tx ": "Carol pays Donna 10 DSCoin",

"PrevHash": "000018785b95a3fa19347207fb85c58599bc2ee3ce7e90fbe771293dcf53490c",

"nonce": "50034",

"difficulty": 4

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

}

],

"chainHash":
"0000175c9f913eb1846c65afae7c8c02359cce218f403a0e0424182af341a299"

}

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

2

Sent:
{"request":"verify","difficulty":0,"blockID":0,"clientID":"ba77d68adec2806639c519a16341e7621b0dac357be3e11169ef","signature":"40544380728139635361021538901581245649035680895365917143489617825019856992406559541632287065787170827096489475961179033408707121700553919395836431213171798883584402948365480913962806665284425077006475577467198066538488639764494090226979990 73","e":65537,"n":448874687648573780753681236128106014244145624770942349574946099856524338242562799230182230664951690337026670615421769641635346658397432304304002590759683036458351592105322698809372417268313551919408060699332727705174541898691644954107401676 1}

Response:

Verifying entire chain

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

Chain verification: TRUE

Total execution time required to verify the chain was 125456 milliseconds

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

4

Enter block ID of block to corrupt: 2

Enter new data for block 2: Bob pays Tony 30 DSCoin

Sent: {"request":"corrupt","data":"Bob pays Tony 30 DSCoin","difficulty":0,"blockID":2,"clientID":"ba77d68adec2806639c519a16341e7621b0dac357be3e11169ef","signature":"26433412899686207207676288969300396763098993431829124732381252643822636817929712447174771147901959301919468161677614528167958355091911816470803066839520350557224600878052366677363008648971373451971280244235714267545090291758374578306328676888","e":65537,"n":448874687648573780753681236128106014244145624770942349574946099856524338242562799230182230664951690337026670615421769641635346658397432304304002590759683036458351592105322698809372417268313551919408060699332727705174541898691644954107401676 1}

Response:

Block 2 now holds: Bob pays Tony 30 DSCoin

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

3

Sent:

{"request":"view","difficulty":0,"blockID":0,"clientID":"ba77d68adec2806639c519a16341e7621b0dac357be3e11169ef","signature":"100612469282391272199364060057076360841211832434181276879525607272325204190045504460333183689729517300216717745532386862863325272188171232462415097162372993227621610573404814521623765529127691213879394424752744452093098336144684787719957240 4","e":65537,"n":448874687648573780753681236128106014244145624770942349574946099856524338242562799230182230664951690337026670615421769641635346658397432304304002590759683036458351592105322698809372417268313551919408060699332727705174541898691644954107401676 1}

Response:

View the Blockchain

{

"ds_chain": [

{

"index": 0,

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

"time stamp ": "2025-03-18 02:39:57.743",

"Tx ": "Genesis",

"PrevHash": "",

"nonce": "246",

"difficulty": 2

},

{

"index": 1,

"time stamp ": "2025-03-18 02:40:24.637",

"Tx ": "Alice pays Bob 100 DSCoin",

"PrevHash":
"00de2df1f886a84cb09e7f118e9772a19b29ac2e1666bdf132f2d86909b1e4a7",

"nonce": "69462",

"difficulty": 4

},

{

"index": 2,

"time stamp ": "2025-03-18 02:40:33.492",

"Tx ": "Bob pays Tony 30 DSCoin",

"PrevHash":
"0000a6363742fe2ec6dde1e9c5adbbcbfc04c4fbffe15dbbedc83d93cfd6f7c7",

"nonce": "50929",

"difficulty": 4

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

},

{

"index": 3,

"time stamp ": "2025-03-18 02:40:41.072",

"Tx ": "Carol pays Donna 10 DSCoin",

"PrevHash":
"000078b73644bccd2b8eabad72397900e6465628f1b3f7eeda0ac97d720e597c",

"nonce": "36221",

"difficulty": 4

},

{

"index": 4,

"time stamp ": "2025-03-18 02:41:45.381",

"Tx ": "Alice pays Bob 100 DSCoin",

"PrevHash":
"f54354c322c1649a098978d8b9a63530ad6a1bb58adc74fe32403a0a5e76a127",

"nonce": "108057",

"difficulty": 4

},

{

"index": 5,

"time stamp ": "2025-03-18 02:41:53.142",

"Tx ": "Bob pays Carol 20 DSCoin",

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

"PrevHash": "0000ed0c56e5a5addefc89f54d83bbc0151faf5884e43480e99f3d6369f561f0",

"nonce": "31620",

"difficulty": 4

},

{

"index": 6,

"time stamp ": "2025-03-18 02:42:00.613",

"Tx ": "Carol pays Donna 10 DSCoin",

"PrevHash": "000018785b95a3fa19347207fb85c58599bc2ee3ce7e90fbe771293dcf53490c",

"nonce": "50034",

"difficulty": 4

}

],

"chainHash": "0000175c9f913eb1846c65afae7c8c02359cce218f403a0e0424182af341a299"

}

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

5. Hide the corruption by repairing the chain.

6. Exit.

2

Sent:

{"request":"verify","difficulty":0,"blockID":0,"clientID":"ba77d68adec2806639c519a16341e7621b0dac357be3e11169ef","signature":"40544380728139635361021538901581245649035680895365917143489617825019856992406559541632287065787170827096489475961179033408707121700553919395836431213171798883584402948365480913962806665284425077006475577467198066538488639764494090226979990 73","e":65537,"n":448874687648573780753681236128106014244145624770942349574946099856524338242562799230182230664951690337026670615421769641635346658397432304304002590759683036458351592105322698809372417268313551919408060699332727705174541898691644954107401676 1}

Response:

Verifying entire chain

Chain verification: TRUE

Total execution time required to verify the chain was 137804 milliseconds

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

5

Sent:

{"request":"hide","difficulty":0,"blockID":0,"clientID":"ba77d68adec2806639c519a16341e7621b0dac357be3e11169ef","signature":"3197722860581965097921921713581418629526943938316046624103734360329285606315417113265264558620894997097003959054967289095810057947252451792650063677142068163248129517975690012747751221885870911123737164146378570111364852884367300489609662669","e":65537,"n":4488746876485737807536812361281060142441456247709423495749460998565243382425627992301822306649516903370266706154217696416353466583974323043040025907596830364583515921053226988093724172683135519194080606993327277051745418986916449541074016761}

Response:

Repairing the entire chain

Total execution time required to repair the chain was 139863 milliseconds

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

2

Sent:

{"request":"verify","difficulty":0,"blockID":0,"clientID":"ba77d68adec2806639c519a16341e7621b0dac357be3e11169ef","signature":"405443807281396353610215 38

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

90158124564903568089536591714348961782501985699240655954163228706578717082709648947596117903340870712170055391939583643121317179888358440294836548091396280666528442507700647557746719806653848863976449409022697999073","e":65537,"n":44887468764857378075368123612810601424414562477094234957494609985652433824256279923018223066495169033702667061542176964163534665839743230430400259075968303645835159210532269880937241726831355191940806069933272770517454189869164495410740167 61}

Response:

Verifying entire chain

Chain verification: TRUE

Total execution time required to verify the chain was 144864 milliseconds

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

3

Sent:

{"request":"view","difficulty":0,"blockID":0,"clientID":"ba77d68adec2806639c519a16341e7621b0dac357be3e11169ef","signature":"100612469282391272199364060057076360841211832434181276879525607272325204190045504460333183689729517300216717745532386862863325272188171232462415097162372993227621610573404814521623765529127691213879394424752744452093098336144684780

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

77199572404","e":65537,"n":448874687648573780753681236128106014244145624770942349574946099856524338242562799230182230664951690337026670615421769641635346658397432304304002590759683036458351592105322698809372417268313551919408060699332727705174541898691644954107401676 1}

Response:

View the Blockchain


```
{
   "ds_chain": [
      {
         "index": 0,
         "time stamp ": "2025-03-18 02:39:57.743",
         "Tx ": "Genesis",
         "PrevHash": "",
         "nonce": "246",
         "difficulty": 2
      },
      {
         "index": 1,
         "time stamp ": "2025-03-18 02:40:24.637",
         "Tx ": "Alice pays Bob 100 DSCoin",
         "PrevHash":
"00de2df1f886a84cb09e7f118e9772a19b29ac2e1666bdf132f2d86909b1e4a7",
         "nonce": "69462",
         "difficulty": 4
```

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

},

{

"index": 2,

"time stamp ": "2025-03-18 02:40:33.492",

"Tx ": "Bob pays Tony 30 DSCoin",

"PrevHash":
"0000a6363742fe2ec6dde1e9c5adbbcbfc04c4fbffe15dbbedc83d93cfd6f7c7",

"nonce": "50929",

"difficulty": 4

},

{

"index": 3,

"time stamp ": "2025-03-18 02:40:41.072",

"Tx ": "Carol pays Donna 10 DSCoin",

"PrevHash":
"000078b73644bccd2b8eabad72397900e6465628f1b3f7eeda0ac97d720e597c",

"nonce": "121961",

"difficulty": 4

},

{

"index": 4,

"time stamp ": "2025-03-18 02:41:45.381",

"Tx ": "Alice pays Bob 100 DSCoin",

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

"PrevHash": "000092ca2831aa8be048b1678a3d3b740fab93e08fb63d42318e84e6f917edd1",

"nonce": "108057",

"difficulty": 4

},

{

"index": 5,

"time stamp ": "2025-03-18 02:41:53.142",

"Tx ": "Bob pays Carol 20 DSCoin",

"PrevHash": "1a8311788a5564cf2fab754c163df704d59d9cf556a49ac4e45d9d53ba83d950",

"nonce": "31620",

"difficulty": 4

},

{

"index": 6,

"time stamp ": "2025-03-18 02:42:00.613",

"Tx ": "Carol pays Donna 10 DSCoin",

"PrevHash": "9aa5c8f3f9230fa199c105165ec725420d09d72823cbd123b89100fa1134977c",

"nonce": "50034",

"difficulty": 4

}

],

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

"chainHash":
"5eca3b5f9fc42bb62ec2ffc3724c8ccbbe887979a2c955ef148796ff8cfeb250"

}

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

6

Process finished with exit code 0

Client 3:

C:\Users\USER\.jdks\openjdk-22.0.2\bin\java.exe "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA 2024.2.0.2\lib\idea_rt.jar=53729:C:\Program Files\JetBrains\IntelliJ IDEA 2024.2.0.2\bin" -Dfile.encoding=UTF-8 -Dsun.stdout.encoding=UTF-8 -Dsun.stderr.encoding=UTF-8 -classpath C:\Users\USER\IdeaProjects\Project3Task2\target\classes;C:\Users\USER\.m2\repository\com\google\code\gson\gson\2.9.0\gson-2.9.0.jar ds.SigningClientTCP3

Client Public Key: e = 65537, n = 191364180181617555234845433888029827911347020867395118901325363421141863134520981393594681368791667568349571918009177710715002424825499 3

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

855936906771431128875696088043765047355370125506199713247084538800450562748612149010415343745218440305663

Client Private Key: d = 130760802552652659696459224871046153371773842081323347340268129830867676501672621401162046188522948213159444735527823919752685457671304606847662367846343823183230136559197018177350723934264102260788517282847755863761670523632748669107868571 3, n = 191364180181617555234845433888029827911347020867395118901325363421141863134520981393594681368791667568349571918009177710715002424825499385593690677143112887569608804376504735537012550619971324708453880045056274861214901041534374521844030566 3

Client ID: eece460e97776379700861e305e990dfc6290f7b43fa962eead4

Client is running...

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

0

Sent:

{"request":"status","difficulty":0,"blockID":0,"clientID":"eece460e9777637970086 1e305e990dfc6290f7b43fa962eead4","signature":"4543267095302432051778528800508713779749491025880902929893761624700374454690317587306051234678088464462500604587831990589836216166025777326314223671423302662003010541581484789120369109155728460299734905588654636693217629591752886 8

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

9256087718","e":65537,"n":191364180181617555234845433888029827911347020867395118901325363421141863134520981393594681368791667568349571918009177710715002424825499385593690677143112887569608804376504735537012550619971324708453880045056274861214901041534374521844030566 3}

Response:

Current size of chain: 7

Difficulty of most recent block: 4

Total difficulty for all blocks: 26

Approximate hashes per second: 0

Expected total hashes required: 393472.0

Nonce for most recent block: 50034

Chain hash:
5eca3b5f9fc42bb62ec2ffc3724c8ccbbe887979a2c955ef148796ff8cfeb250

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

1

Enter difficulty > 1: 4

Enter transaction: Alice pays Bob 100 DSCoin

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

Sent: {"request":"add transaction","data":"Alice pays Bob 100 DSCoin","difficulty":4,"blockID":0,"clientID":"eece460e97776379700861e305e990dfc6290f7b43fa962eead4","signature":"1149021273782741613077444946530275169149247424822760825758438570613772855811051338068456675928876790771465975953033806813939672927834298273911948441541782720028095021162706712945073559615972532046875542809688260949309419892735314125914709340","e":65537,"n":19136418018161755523484543388802982791134702086739511890132536342114186313452098139359468136879166756834957191800917771071500242482549938559369067714311288756960880437650473553701255061997132470845388004505627486121490104153437452184403056630}

Response:

Total execution time to add this block was 165517 milliseconds

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

1

Enter difficulty > 1: 4

Enter transaction: Bob pays Carol 20 DSCoin

Sent: {"request":"add transaction","data":"Bob pays Carol 20 DSCoin","difficulty":4,"blockID":0,"clientID":"eece460e97776379700861e305e990dfc6290f7b43fa962eead4","signature":"1347157661435766097324033619622170l

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

312126198374392672296395854945539430735825888581911718321788246286118878028165117256422314341594347452781670196715376627202698199369938630874582919617211222248544583241490211553628251954944003746441153582
62","e":65537,"n":191364180181617555234845433888029827911347020867395118901325363421141863134520981393594681368791667568349571918009177710715002424825499385593690677143112887569608804376504735537012550619971324708453880045056274861214901041534374521844030566 3}

Response:

Total execution time to add this block was 175728 milliseconds

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

1

Enter difficulty > 1: 4

Enter transaction: Carol pays Donna 10 DSCoin

Sent: {"request":"add transaction","data":"Carol pays Donna 10 DSCoin","difficulty":4,"blockID":0,"clientID":"eece460e97776379700861e305e990dfc6290f7b43fa962eead4","signature":"182273793030186651605589927229871705090397762502537425565448354528537024564578553682737868466889918967197419584268322402046981405905436528538823269026319004349429992099513637962891067621419262537567331699084457266339926171138787911831441636

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

07","e":65537,"n":191364180181617555234845433888029827911347020867395118901325363421141863134520981393594681368791667568349571918009177107150024248254993855936906771431128875696088043765047355370125506199713247084538800450562748612149010415343745218440305663}

Response:

Total execution time to add this block was 181314 milliseconds

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

3

Sent:

{"request":"view","difficulty":0,"blockID":0,"clientID":"eece460e97776379700861e305e990dfc6290f7b43fa962eead4","signature":"189929307541776555105565110570924001270251310255100304304205640681794646694716969316296759353977414513330105123908838005800057401021050742631653381129167253959575989271074203889658912241256323186944179655035727850301233624674786191026212332 6","e":65537,"n":191364180181617555234845433888029827911347020867395118901325363421141863134520981393594681368791667568349571918009177107150024248254993855936906771431128875696088043765047355370125506199713247084538800450562748612149010415343745218440305663}

Response:

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

View the Blockchain

{

"ds_chain": [

{

"index": 0,

"time stamp ": "2025-03-18 02:39:57.743",

"Tx ": "Genesis",

"PrevHash": "",

"nonce": "246",

"difficulty": 2

},

{

"index": 1,

"time stamp ": "2025-03-18 02:40:24.637",

"Tx ": "Alice pays Bob 100 DSCoin",

"PrevHash":
"00de2df1f886a84cb09e7f118e9772a19b29ac2e1666bdf132f2d86909b1e4a7",

"nonce": "69462",

"difficulty": 4

},

{

"index": 2,

"time stamp ": "2025-03-18 02:40:33.492",

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

"Tx ": "Bob pays Tony 30 DSCoin",

"PrevHash": "0000a6363742fe2ec6dde1e9c5adbbcbfc04c4fbffe15dbbedc83d93cfd6f7c7",

"nonce": "50929",

"difficulty": 4

},

{

"index": 3,

"time stamp ": "2025-03-18 02:40:41.072",

"Tx ": "Carol pays Donna 10 DSCoin",

"PrevHash": "000078b73644bccd2b8eabad72397900e6465628f1b3f7eeda0ac97d720e597c",

"nonce": "121961",

"difficulty": 4

},

{

"index": 4,

"time stamp ": "2025-03-18 02:41:45.381",

"Tx ": "Alice pays Bob 100 DSCoin",

"PrevHash": "000092ca2831aa8be048b1678a3d3b740fab93e08fb63d42318e84e6f917edd1",

"nonce": "108057",

"difficulty": 4

},

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

{

"index": 5,

"time stamp ": "2025-03-18 02:41:53.142",

"Tx ": "Bob pays Carol 20 DSCoin",

"PrevHash": "1a8311788a5564cf2fab754c163df704d59d9cf556a49ac4e45d9d53ba83d950",

"nonce": "31620",

"difficulty": 4

},

{

"index": 6,

"time stamp ": "2025-03-18 02:42:00.613",

"Tx ": "Carol pays Donna 10 DSCoin",

"PrevHash": "9aa5c8f3f9230fa199c105165ec725420d09d72823cbd123b89100fa1134977c",

"nonce": "50034",

"difficulty": 4

},

{

"index": 7,

"time stamp ": "2025-03-18 02:42:48.369",

"Tx ": "Alice pays Bob 100 DSCoin",

"PrevHash": "5eca3b5f9fc42bb62ec2ffc3724c8ccbbe887979a2c955ef148796ff8cfeb250",

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

"nonce": "46870",

"difficulty": 4

},

{

"index": 8,

"time stamp ": "2025-03-18 02:42:58.63",

"Tx ": "Bob pays Carol 20 DSCoin",

"PrevHash":
"000056d586208a739068dbcbf457feb9ee909e5586d9c5bb04244e50d12a2aa1",

"nonce": "16300",

"difficulty": 4

},

{

"index": 9,

"time stamp ": "2025-03-18 02:43:04.202",

"Tx ": "Carol pays Donna 10 DSCoin",

"PrevHash":
"00005765b1bafba6721013b44ba234b8356efaf31c89d5c76d76a019983ffd11",

"nonce": "25438",

"difficulty": 4

}

],

"chainHash":
"0000fea1d82e87157a0096546801034429e4c1de50b177364831ffc8706743c2"

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

}

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

2

Sent:

{"request":"verify","difficulty":0,"blockID":0,"clientID":"eece460e97776379700861e305e990dfc6290f7b43fa962eead4","signature":"184574394548705401035465542470868194396360792775369371497006764740227804098133241131865028508076525572963645403827982081340937006296284865417016183260528116993191010518016354877047385803196349447223943298702298100940603899593074827026049028 1","e":65537,"n":191364180181617555234845433888029827911347020867395118901325363421141863134520981393594681368791667568349571918009177710715002424825499385593690677143112887569608804376504735537012550619971324708453880045056274861214901041534374521844030566 3}

Response:

Verifying entire chain

Chain verification: TRUE

Total execution time required to verify the chain was 187972 milliseconds

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

4

Enter block ID of block to corrupt: 2

Enter new data for block 2: Bob pays Tony 30 DSCoin

Sent: {"request":"corrupt","data":"Bob pays Tony 30 DSCoin","difficulty":0,"blockID":2,"clientID":"eece460e97776379700861e305e990dfc6290f7b43fa962eead4","signature":"820915406833011136860143949479458779024552933345681050702134019788409941739546323821388327824252352048785302772951462506586601296737940027861015605361039666109901115735569775822559424890022946799427698940881377683304839812429827784546564807","e":65537,"n":19136418018161755523484543388802982791134702086739511890132536342114186313452098139359468136879166756834957191800917771071500242482549938559369067714311288756960880437650473553701255061997132470845388004505627486121490104153437452184403056 63}

Response:

Block 2 now holds: Bob pays Tony 30 DSCoin

0. View basic blockchain status.

1. Add a transaction to the blockchain.

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

3

Sent:

{"request":"view","difficulty":0,"blockID":0,"clientID":"eece460e97776379700861e305e990dfc6290f7b43fa962eead4","signature":"18992930754177655510556511057092400127025131025510030430420564068179464669471696931629675935397741451333010512390883800580005740102105074263165338112916725395957598927107420388965891224125632318694417965503572785030123362467478619102621233 26","e":65537,"n":19136418018161755523484543388802982791134702086739511890132536342114186313452098139359468136879166756834957191800917771071500242482549938559369067714311288756960880437650473553701255061997132470845388004505627486121490104153437452184403056 63}

Response:

View the Blockchain

{

"ds_chain": [

{

"index": 0,

"time stamp ": "2025-03-18 02:39:57.743",

"Tx ": "Genesis",

"PrevHash": "",

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

"nonce": "246",

"difficulty": 2

},

{

"index": 1,

"time stamp ": "2025-03-18 02:40:24.637",

"Tx ": "Alice pays Bob 100 DSCoin",

"PrevHash":
"00de2df1f886a84cb09e7f118e9772a19b29ac2e1666bdf132f2d86909b1e4a7",

"nonce": "69462",

"difficulty": 4

},

{

"index": 2,

"time stamp ": "2025-03-18 02:40:33.492",

"Tx ": "Bob pays Tony 30 DSCoin",

"PrevHash":
"0000a6363742fe2ec6dde1e9c5adbbcbfc04c4fbffe15dbbedc83d93cfd6f7c7",

"nonce": "50929",

"difficulty": 4

},

{

"index": 3,

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

"time stamp ": "2025-03-18 02:40:41.072",

"Tx ": "Carol pays Donna 10 DSCoin",

"PrevHash": "000078b73644bccd2b8eabad72397900e6465628f1b3f7eeda0ac97d720e597c",

"nonce": "121961",

"difficulty": 4

},

{

"index": 4,

"time stamp ": "2025-03-18 02:41:45.381",

"Tx ": "Alice pays Bob 100 DSCoin",

"PrevHash": "000092ca2831aa8be048b1678a3d3b740fab93e08fb63d42318e84e6f917edd1",

"nonce": "108057",

"difficulty": 4

},

{

"index": 5,

"time stamp ": "2025-03-18 02:41:53.142",

"Tx ": "Bob pays Carol 20 DSCoin",

"PrevHash": "1a8311788a5564cf2fab754c163df704d59d9cf556a49ac4e45d9d53ba83d950",

"nonce": "31620",

"difficulty": 4

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

},

{

"index": 6,

"time stamp ": "2025-03-18 02:42:00.613",

"Tx ": "Carol pays Donna 10 DSCoin",

"PrevHash": "9aa5c8f3f9230fa199c105165ec725420d09d72823cbd123b89100fa1134977c",

"nonce": "50034",

"difficulty": 4

},

{

"index": 7,

"time stamp ": "2025-03-18 02:42:48.369",

"Tx ": "Alice pays Bob 100 DSCoin",

"PrevHash": "5eca3b5f9fc42bb62ec2ffc3724c8ccbbe887979a2c955ef148796ff8cfeb250",

"nonce": "46870",

"difficulty": 4

},

{

"index": 8,

"time stamp ": "2025-03-18 02:42:58.63",

"Tx ": "Bob pays Carol 20 DSCoin",

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

"PrevHash": "000056d586208a739068dbcbf457feb9ee909e5586d9c5bb04244e50d12a2aa1",

"nonce": "16300",

"difficulty": 4

},

{

"index": 9,

"time stamp ": "2025-03-18 02:43:04.202",

"Tx ": "Carol pays Donna 10 DSCoin",

"PrevHash": "00005765b1bafba6721013b44ba234b8356efaf31c89d5c76d76a019983ffd11",

"nonce": "25438",

"difficulty": 4

}

],

"chainHash": "0000fea1d82e87157a0096546801034429e4c1de50b177364831ffc8706743c2"

}

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

5. Hide the corruption by repairing the chain.

6. Exit.

2

Sent:

{"request":"verify","difficulty":0,"blockID":0,"clientID":"eece460e97776379700861e305e990dfc6290f7b43fa962eead4","signature":"184574394548705401035465542470868194396360792775369371497006764740227804098133241131865028508076525572963645403827982081340937006296284865417016183260528116993191010518016354877047385803196349447223943298702298100940603899593074827026049028 1","e":65537,"n":191364180181617555234845433888029827911347020867395118901325363421141863134520981393594681368791667568349571918009177710715002424825499385593690677143112887569608804376504735537012550619971324708453880045056274861214901041534374521844030566 3}

Response:

Verifying entire chain

Chain verification: TRUE

Total execution time required to verify the chain was 199971 milliseconds

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

5

Sent:

{"request":"hide","difficulty":0,"blockID":0,"clientID":"eece460e97776379700861e305e990dfc6290f7b43fa962eead4","signature":"290509127954117379685723765852352080812457342811196939333418269507935081255075763650246413690770341798349977903379095625330339296029715332795765990172685923570625756963068250407393760289302867759178305395923955530693059004774997671378870108","e":65537,"n":191364180181617555234845433888029827911347020867395118901325363421141863134520981393594681368791667568349571918009177710715002424825499385593690677143112887569608804376504735537012550619971324708453880045056274861214901041534374521844030566 3}

Response:

Repairing the entire chain

Total execution time required to repair the chain was 209006 milliseconds

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

2

Sent:

{"request":"verify","difficulty":0,"blockID":0,"clientID":"eece460e97776379700861e305e990dfc6290f7b43fa962eead4","signature":"184574394548705401035465542

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

4708681943963607927753693714970067647402278040981332411318650285080765255729636454038279820813409370062962848654170161832605281169931910105180163548770473858031963494472239432987022981009406038995930748270260490281","e":65537,"n":191364180181617555234845433888029827911347020867395118901325363421141863134520981393594681368791667568349571918009177710715002424825499385593690677143112887569608804376504735537012550619971324708453880045056274861214901041534374521844030566}

Response:

Verifying entire chain

Chain verification: TRUE

Total execution time required to verify the chain was 215813 milliseconds

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

3

Sent:

{"request":"view","difficulty":0,"blockID":0,"clientID":"eece460e97776379700861e305e990dfc6290f7b43fa962eead4","signature":"189929307541776555105565110570924001270251310255100304304205640681794646694716969316296759353977414513330105123908838005800057401021050742631653381129167253959575989271074203889658912241256323186944179655035727850301233624674786191

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

0262123326","e":65537,"n":191364180181617555234845433888029827911347020867395118901325363421141863134520981393594681368791667568349571918009177710715002424825499385593690677143112887569608804376504735537012550619971324708453880045056274861214901041534374521844030566 3}

Response:

View the Blockchain

{

"ds_chain": [

{

"index": 0,

"time stamp ": "2025-03-18 02:39:57.743",

"Tx ": "Genesis",

"PrevHash": "",

"nonce": "246",

"difficulty": 2

},

{

"index": 1,

"time stamp ": "2025-03-18 02:40:24.637",

"Tx ": "Alice pays Bob 100 DSCoin",

"PrevHash":
"00de2df1f886a84cb09e7f118e9772a19b29ac2e1666bdf132f2d86909b1e4a7",

"nonce": "69462",

"difficulty": 4

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

},

{

"index": 2,

"time stamp ": "2025-03-18 02:40:33.492",

"Tx ": "Bob pays Tony 30 DSCoin",

"PrevHash":
"0000a6363742fe2ec6dde1e9c5adbbcbfc04c4fbffe15dbbedc83d93cfd6f7c7",

"nonce": "50929",

"difficulty": 4

},

{

"index": 3,

"time stamp ": "2025-03-18 02:40:41.072",

"Tx ": "Carol pays Donna 10 DSCoin",

"PrevHash":
"000078b73644bccd2b8eabad72397900e6465628f1b3f7eeda0ac97d720e597c",

"nonce": "121961",

"difficulty": 4

},

{

"index": 4,

"time stamp ": "2025-03-18 02:41:45.381",

"Tx ": "Alice pays Bob 100 DSCoin",

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

"PrevHash": "000092ca2831aa8be048b1678a3d3b740fab93e08fb63d42318e84e6f917edd1",

"nonce": "194450",

"difficulty": 4

},

{

"index": 5,

"time stamp ": "2025-03-18 02:41:53.142",

"Tx ": "Bob pays Carol 20 DSCoin",

"PrevHash": "00000b2b67dae3ecba5cfa5427b79ee47f85c362bede1f3a128fd2490c5aa069",

"nonce": "249128",

"difficulty": 4

},

{

"index": 6,

"time stamp ": "2025-03-18 02:42:00.613",

"Tx ": "Carol pays Donna 10 DSCoin",

"PrevHash": "c5eb2ba7292578fa471e9f7c92c8de5d53c112d81aa6627e0552075bd2cbd422",

"nonce": "89065",

"difficulty": 4

},

{

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

"index": 7,

"time stamp ": "2025-03-18 02:42:48.369",

"Tx ": "Alice pays Bob 100 DSCoin",

"PrevHash": "72e7901d3ede52a3bc31be2b13536e2d311c2e691576de78b66662c544e5d1dd",

"nonce": "46870",

"difficulty": 4

},

{

"index": 8,

"time stamp ": "2025-03-18 02:42:58.63",

"Tx ": "Bob pays Carol 20 DSCoin",

"PrevHash": "1794a86aa4f794eba0baf1ae6b80eb607a83c97645f09357763223513162d9d3",

"nonce": "16300",

"difficulty": 4

},

{

"index": 9,

"time stamp ": "2025-03-18 02:43:04.202",

"Tx ": "Carol pays Donna 10 DSCoin",

"PrevHash": "43c2f6654a3e83e7bdb00f4582dec6ccecf222033379d30269e96f4d4523ec72",

"nonce": "25438",

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

"difficulty": 4

}

],

"chainHash":
"7e0394a2848727eb8f665fdc7e7b8a13398b2345ffdb9d993ebf078972047fc5"

}

0. View basic blockchain status.

1. Add a transaction to the blockchain.

2. Verify the blockchain.

3. View the blockchain.

4. Corrupt the chain.

5. Hide the corruption by repairing the chain.

6. Exit.

6

Process finished with exit code 0

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

ServerOutput:

C:\Users\USER\.jdks\openjdk-22.0.2\bin\java.exe "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA 2024.2.0.2\lib\idea_rt.jar=53726:C:\Program Files\JetBrains\IntelliJ IDEA 2024.2.0.2\bin" -Dfile.encoding=UTF-8 -Dsun.stdout.encoding=UTF-8 -Dsun.stderr.encoding=UTF-8 -classpath C:\Users\USER\IdeaProjects\Project3Task2\target\classes;C:\Users\USER\.m2\repository\com\google\code\gson\gson\2.9.0\gson-2.9.0.jar ds.VerifyingServerTCP

Blockchain server is running...

New client connected...

New client connected...

New client connected...

We have a visitor

THE JSON REQUEST MESSAGE IS SHOWN HERE:

{"request":"status","difficulty":0,"blockID":0,"clientID":"03ee0037c93a333ef371d410cbcb4138d98134772102d51aa41","signature":"434502712166497811938292751693091039265205067982737939863194521120726333314268805162804577930325853804832013127904051158426993680702850026788807690648811507016338785807267057500493383615997058143741637876852683943034707390316457889290165329 3","e":65537,"n":458059865128652378038727029632542768974397082279984574317839934053260764280652910602838420946732019319243824177622799845743178399340532607642806529106028384209467320193192438241776

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

4756597424404181910337959678481787167603542730310931749545380400922730406407814125245487200122387802549181430417991469774833119 53}

Computed hash: 2937557172

Decrypted signature: 2937557172

THE JSON RESPONSE MESSAGE IS SHOWN HERE:

{"status":"success","message":"Current size of chain: 1\nDifficulty of most recent block: 2\nTotal difficulty for all blocks: 2\nApproximate hashes per second: 0\nExpected total hashes required: 256.0\nNonce for most recent block: 246\nChain hash: 00de2df1f886a84cb09e7f118e9772a19b29ac2e1666bdf132f2d86909b1e4a7\n"}

Number of Blocks on Chain == 1

We have a visitor

THE JSON REQUEST MESSAGE IS SHOWN HERE:

{"request":"add transaction","data":"Alice pays Bob 100 DSCoin","difficulty":4,"blockID":0,"clientID":"03ee0037c93a333ef371d410cbcb4138d98134772102d51aa41","signature":"3030889996907864559368382845503334730033155737093380542818169152225411602244280214825952103607433074052562037302864743693170529459171344808253495883239779676146209816901477003768026797388093729290393034321227327345712445390616993524365714636","e":65537,"n":458059865128652378038727029632542768974397087227998457431783993405326076428065291060283842094673201931924382417764756597424404181910337959678481787167603542730310931749545380400922730406407814125245487200122387802549181430417991469774833119 53}

Computed hash: 1292194266

Decrypted signature: 1292194266

THE JSON RESPONSE MESSAGE IS SHOWN HERE:

{"status":"success","message":"Total execution time to add this block was 18343

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

milliseconds\n"}

Number of Blocks on Chain == 2

We have a visitor

THE JSON REQUEST MESSAGE IS SHOWN HERE:

{"request":"add transaction","data":"Bob pays Carol 20 DSCoin","difficulty":4,"blockID":0,"clientID":"03ee0037c93a333ef371d410cbcb4138d98134772102d51aa41","signature":"353407444382347622669064705149296240629519593469819885613940287341724406353788480099121957176838322400598202539484923304305910361485839678374914489470595567969741824185727210665736212190451443542018062792108396838645876356232417003364848376","e":65537,"n":458059865128652378038727029632542768974397082279984574317839934053260764280652910602838420946732019319243824177647565974244041819103379596784817871676035427303109317495453804009227304064078141252454872001223878025491814304179914697748331195 3}

Computed hash: 873736688

Decrypted signature: 873736688

THE JSON RESPONSE MESSAGE IS SHOWN HERE:

{"status":"success","message":"Total execution time to add this block was 26838 milliseconds\n"}

Number of Blocks on Chain == 3

We have a visitor

THE JSON REQUEST MESSAGE IS SHOWN HERE:

{"request":"add transaction","data":"Carol pays Donna 10 DSCoin","difficulty":4,"blockID":0,"clientID":"03ee0037c93a333ef371d410cbcb4138d98134772102d51aa41","signature":"394850918089711359433536025545887008711660546271615722258414723654214441228786691328732422654047004582215035698175968594126573758615230304527794943361760171321146330337346 8

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

5291303257644559671912211330055913703914059053552248798634237867661 9","e":65537,"n":458059865128652378038727029632542768974397082279984574317839934053260764280652910602838420946732019319243824177647565974244041819103379596784817871676035427303109317495453804009227304064078141252454872001223878025491814304179914697748331195 3}

Computed hash: 1780477582

Decrypted signature: 1780477582

THE JSON RESPONSE MESSAGE IS SHOWN HERE:

{"status":"success","message":"Total execution time to add this block was 34499 milliseconds\n"}

Number of Blocks on Chain == 4

We have a visitor

THE JSON REQUEST MESSAGE IS SHOWN HERE:

{"request":"view","difficulty":0,"blockID":0,"clientID":"03ee0037c93a333ef371d410cbcb4138d98134772102d51aa41","signature":"28946006605185601207674322671543986570718959539189529655476145626246294417872131007602963525899892606943116847430100758226721461398914832578240621801887490772520716324803724580804707216654430291045844177225995120950629169434723658509123667 21","e":65537,"n":458059865128652378038727029632542768974397082279984574317839934053260764280652910602838420946732019319243824177647565974244041819103379596784817871676035427303109317495453804009227304064078141252454872001223878025491814304179914697748331195 3}

Computed hash: 2481487641

Decrypted signature: 2481487641

THE JSON RESPONSE MESSAGE IS SHOWN HERE:

{"status":"success","message":"View the Blockchain\n{\n \"ds_chain\": [\n {\n \"index\": 0,\n \"time stamp \": \"2025-03-18 02:39:57.743\",\n

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

\"Tx \": \"Genesis\",\n \"PrevHash\": \"\",\n \"nonce\": \"246\",\n \"difficulty\": 2\n },\n {\n \"index\": 1,\n \"time stamp \": \"2025-03-18 02:40:24.637\",\n \"Tx \": \"Alice pays Bob 100 DSCoin\",\n \"PrevHash\": \"00de2df1f886a84cb09e7f118e9772a19b29ac2e1666bdf132f2d86909b1e4a7\",\n \"nonce\": \"69462\",\n \"difficulty\": 4\n },\n {\n \"index\": 2,\n \"time stamp \": \"2025-03-18 02:40:33.492\",\n \"Tx \": \"Bob pays Carol 20 DSCoin\",\n \"PrevHash\": \"0000a6363742fe2ec6dde1e9c5adbbcbfc04c4fbffe15dbbedc83d93cfd6f7c7\",\n \"nonce\": \"3899\",\n \"difficulty\": 4\n },\n {\n \"index\": 3,\n \"time stamp \": \"2025-03-18 02:40:41.072\",\n \"Tx \": \"Carol pays Donna 10 DSCoin\",\n \"PrevHash\": \"000056dc8a4ba49c9133527f5fdfd7ebf8a11b1bc48562aaa932a514ef60bce5\",\n \"nonce\": \"36221\",\n \"difficulty\": 4\n }\n],\n \"chainHash\": \"0000c6e10ef23ebd325563f24e69ae0b17ee3759773a6119c06d308446834f1a\"\n }\n"}

Number of Blocks on Chain == 4

We have a visitor

THE JSON REQUEST MESSAGE IS SHOWN HERE:

{"request":"verify","difficulty":0,"blockID":0,"clientID":"03ee0037c93a333ef371d410cbcb4138d98134772102d51aa41","signature":"167109993880994211425866479511788699395041413315232560499141595618716662770506551990986786106673981523336484197985201658607296764988030677441625231837537679560536328472098221031480447210838207209347990894068066995375964007914861635411348922","e":65537,"n":458059865128652378038727029632542768974397082279984574317839934053260764280652910602838420946732019319243824177647565974244041819103379596784817871676035427303109317495453804009227304064078141252454872001223878025491814304179914697748331195 3}

Computed hash: 1451567433

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

Decrypted signature: 1451567433

THE JSON RESPONSE MESSAGE IS SHOWN HERE:

{"status":"success","message":"Verifying entire chain\nChain verification: TRUE\nTotal execution time required to verify the chain was 39775 milliseconds\n"}

Number of Blocks on Chain == 4

We have a visitor

THE JSON REQUEST MESSAGE IS SHOWN HERE:

{"request":"corrupt","data":"Bob pays Tony 30 DSCoin","difficulty":0,"blockID":2,"clientID":"03ee0037c93a333ef371d410cbcb4138d98134772102d51aa41","signature":"829673892454249220548004090798303449767179547681425742964034493768745066458147141254641202426423887155426069480081437703500058780985177130829029240585948806166521022445820345018275069176151395488212277280306732479141853702232588205834945822","e":65537,"n":458059865128652378038727029632542768974397082279984574317839934053260764280652910602838420946732019319243824177647565974244041819103379596784817871676035427303109317495453804009227304064078141252454872001223878025491814304179914697748331195 3}

Computed hash: 1179454504

Decrypted signature: 1179454504

THE JSON RESPONSE MESSAGE IS SHOWN HERE:

{"status":"success","message":"Block 2 now holds: Bob pays Tony 30 DSCoin\n"}

Number of Blocks on Chain == 4

We have a visitor

THE JSON REQUEST MESSAGE IS SHOWN HERE:

{"request":"view","difficulty":0,"blockID":0,"clientID":"03ee0037c93a333ef371d41

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

0cbcb4138d98134772102d51aa41","signature":"28946006605185601207674322671543986570718959539189529655476145626246294417872131007602963525899892606943116847430100758226721461398914832578240621801887490772520716324803724580804707216654430291045844177225995120950629169434723658509123667 21","e":65537,"n":458059865128652378038727029632542768974397082279984574317839934053260764280652910602838420946732019319243824177647565974244041819103379596784817871676035427303109317495453804009227304064078141252454872001223878025491814304179914697748331195 3}

Computed hash: 2481487641

Decrypted signature: 2481487641

THE JSON RESPONSE MESSAGE IS SHOWN HERE:

{"status":"success","message":"View the Blockchain\n{\n \"ds_chain\": [\n {\n \"index\": 0,\n \"time stamp \": \"2025-03-18 02:39:57.743\",\n \"Tx \": \"Genesis\",\n \"PrevHash\": \"\",\n \"nonce\": \"246\",\n \"difficulty\": 2\n },\n {\n \"index\": 1,\n \"time stamp \": \"2025-03-18 02:40:24.637\",\n \"Tx \": \"Alice pays Bob 100 DSCoin\",\n \"PrevHash\": \"00de2df1f886a84cb09e7f118e9772a19b29ac2e1666bdf132f2d86909b1e4a7\",\n \"nonce\": \"69462\",\n \"difficulty\": 4\n },\n {\n \"index\": 2,\n \"time stamp \": \"2025-03-18 02:40:33.492\",\n \"Tx \": \"Bob pays Tony 30 DSCoin\",\n \"PrevHash\": \"0000a6363742fe2ec6dde1e9c5adbbcbfc04c4fbffe15dbbedc83d93cfd6f7c7\",\n \"nonce\": \"3899\",\n \"difficulty\": 4\n },\n {\n \"index\": 3,\n \"time stamp \": \"2025-03-18 02:40:41.072\",\n \"Tx \": \"Carol pays Donna 10 DSCoin\",\n \"PrevHash\": \"000056dc8a4ba49c9133527f5fdfd7ebf8a11b1bc48562aaa932a514ef60bce5\",\n \"nonce\": \"36221\",\n \"difficulty\": 4\n }\n],\n \"chainHash\": \"0000c6e10ef23ebd325563f24e69ae0b17ee3759773a6119c06d308446834f1a\"\n}\n"}

Number of Blocks on Chain == 4

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

We have a visitor

THE JSON REQUEST MESSAGE IS SHOWN HERE:

{"request":"verify","difficulty":0,"blockID":0,"clientID":"03ee0037c93a333ef371d4 10cbcb4138d98134772102d51aa41","signature":"167109993880994211425866479 951178869939504141331523256049914159561871666277050655199098678610667 398152333648419798520165860729676498803067744162523183753767956053632847209822103148044721083820720934799089406806699537596400791486163 5411348922","e":65537,"n":458059865128652378038727029632542768974397082279984574317839934053260764280652910602838420946732019319243824177647565974244041819103379596784817871676035427303109317495453804009227304064078141252454872001223878025491814304179914697748331195 3}

Computed hash: 1451567433

Decrypted signature: 1451567433

THE JSON RESPONSE MESSAGE IS SHOWN HERE:

{"status":"failed","message":"Chain verification: FALSE\nImproper hash on node 2 Does not begin with 0000\n"}

Number of Blocks on Chain == 4

We have a visitor

THE JSON REQUEST MESSAGE IS SHOWN HERE:

{"request":"hide","difficulty":0,"blockID":0,"clientID":"03ee0037c93a333ef371d41 0cbcb4138d98134772102d51aa41","signature":"279672779487993891534673303541874803866841725641062650198373877387582895750811710140260322327884447251397332560097696717175003982068035713121151007468024749272503884736460871520160139564528808028128084794365599421179939944306790382468341400 9","e":65537,"n":458059865128652378038727029632542768974397082279984574317839934053260764280652910602838420946732019319243824177647565974244041819103379596784817871676035427303109317495453804009227304064078141252454872001223878025491814304179914697748331195 3}

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

Computed hash: 3359791462

Decrypted signature: 3359791462

THE JSON RESPONSE MESSAGE IS SHOWN HERE:

{"status":"success","message":"Repairing the entire chain\nTotal execution time required to repair the chain was 53452 milliseconds\n"}

Number of Blocks on Chain == 4

We have a visitor

THE JSON REQUEST MESSAGE IS SHOWN HERE:

{"request":"verify","difficulty":0,"blockID":0,"clientID":"03ee0037c93a333ef371d410cbcb4138d98134772102d51aa41","signature":"167109993880994211425866479951178869939504141331523256049914159561871666277050655199098678610667398152333648419798520165860729676498803067744162523183753767956053632847209822103148044721083820720934799089406806699537596400791486163541134892 2","e":65537,"n":458059865128652378038727029632542768974397082279984574317839934053260764280652910602838420946732019319243824177647565974244041819103379596784817871676035427303109317495453804009227304064078141252454872001223878025491814304179914697748331195 3}

Computed hash: 1451567433

Decrypted signature: 1451567433

THE JSON RESPONSE MESSAGE IS SHOWN HERE:

{"status":"success","message":"Verifying entire chain\nChain verification: TRUE\nTotal execution time required to verify the chain was 58447 milliseconds\n"}

Number of Blocks on Chain == 4

We have a visitor

THE JSON REQUEST MESSAGE IS SHOWN HERE:

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

{"request":"view","difficulty":0,"blockID":0,"clientID":"03ee0037c93a333ef371d410cbcb4138d98134772102d51aa41","signature":"2894600660518560120767432267154398657071895953918952965547614562624629441787213100760296352589989260694311684743010075822672146139891483257824062180188749077252071632480372458080470721665443029104584417722599512095062916943472365850912366721","e":65537,"n":4580598651286523780387270296325427689743970822799845743178399340532607642806529106028384209467320193192438241776475659742440418191033795967848178716760354273031093174954538040092273040640781412524548720012238780254918143041799146977483311953}

Computed hash: 2481487641

Decrypted signature: 2481487641

THE JSON RESPONSE MESSAGE IS SHOWN HERE:

{"status":"success","message":"View the Blockchain\n{\n \"ds_chain\": [\n {\n \"index\": 0,\n \"time stamp \": \"2025-03-18 02:39:57.743\",\n \"Tx \": \"Genesis\",\n \"PrevHash\": \"\",\n \"nonce\": \"246\",\n \"difficulty\": 2\n },\n {\n \"index\": 1,\n \"time stamp \": \"2025-03-18 02:40:24.637\",\n \"Tx \": \"Alice pays Bob 100 DSCoin\",\n \"PrevHash\": \"00de2df1f886a84cb09e7f118e9772a19b29ac2e1666bdf132f2d86909b1e4a7\",\n \"nonce\": \"69462\",\n \"difficulty\": 4\n },\n {\n \"index\": 2,\n \"time stamp \": \"2025-03-18 02:40:33.492\",\n \"Tx \": \"Bob pays Tony 30 DSCoin\",\n \"PrevHash\": \"0000a6363742fe2ec6dde1e9c5adbbcbfc04c4fbffe15dbbedc83d93cfd6f7c7\",\n \"nonce\": \"50929\",\n \"difficulty\": 4\n },\n {\n \"index\": 3,\n \"time stamp \": \"2025-03-18 02:40:41.072\",\n \"Tx \": \"Carol pays Donna 10 DSCoin\",\n \"PrevHash\": \"000078b73644bccd2b8eabad72397900e6465628f1b3f7eeda0ac97d720e597c\",\n \"nonce\": \"36221\",\n \"difficulty\": 4\n }\n],\n \"chainHash\": \"f54354c322c1649a098978d8b9a63530ad6a1bb58adc74fe32403a0a5e76a127\"\n

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

}\n"}

Number of Blocks on Chain == 4

We have a visitor

THE JSON REQUEST MESSAGE IS SHOWN HERE:

{"request":"status","difficulty":0,"blockID":0,"clientID":"ba77d68adec2806639c519a16341e7621b0dac357be3e11169ef","signature":"165244607576019567850556296901625915777689720789251298208998267254273867896381205743465815354803918986041643629672054860723840549637981090540453345800551010214066935217549529450229349732096454932896980996922119221576056538656188381942302541 9","e":65537,"n":448874687648573780753681236128106014244145624770942349574946099856524338242562799230182230664951690337026670615421769641635346658397432304304002590759683036458351592105322698809372417268313551919408060699332727705174541898691644954107401676 1}

Computed hash: 3815041230

Decrypted signature: 3815041230

THE JSON RESPONSE MESSAGE IS SHOWN HERE:

{"status":"success","message":"Current size of chain: 4\nDifficulty of most recent block: 4\nTotal difficulty for all blocks: 14\nApproximate hashes per second: 0\nExpected total hashes required: 196864.0\nNonce for most recent block: 36221\nChain hash: f54354c322c1649a098978d8b9a63530ad6a1bb58adc74fe32403a0a5e76a127\n"}

Number of Blocks on Chain == 4

We have a visitor

THE JSON REQUEST MESSAGE IS SHOWN HERE:

{"request":"add transaction","data":"Alice pays Bob 100 DSCoin","difficulty":4,"blockID":0,"clientID":"ba77d68adec2806639c519a16341e7621b0dac357be3e11169ef","signature":"32276985700204668786784623904663968

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

6373895192337624925257487557344905914558384882201134327337505950241025779078342093687083199757077699500249733624000237081957816427718759718680471783820027323143638502210328721375571553621142534031735425029 4","e":65537,"n":4488746876485737807536812361281060142441456247709423495749460998565243382425627992301822306649516903370266706154217696416353466583974323043040025907596830364583515921053226988093724172683135519194080606993327277051745418986916449541074016761}

Computed hash: 3029608496

Decrypted signature: 3029608496

THE JSON RESPONSE MESSAGE IS SHOWN HERE:

{"status":"success","message":"Total execution time to add this block was 100830 milliseconds\n"}

Number of Blocks on Chain == 5

We have a visitor

THE JSON REQUEST MESSAGE IS SHOWN HERE:

{"request":"add transaction","data":"Bob pays Carol 20 DSCoin","difficulty":4,"blockID":0,"clientID":"ba77d68adec2806639c519a16341e7621b0dac357be3e11169ef","signature":"18614193892217844696529206635708740490290660067773728740359871137149010021745029797234008063206458038230485532481557481845355779534253315086613519885786197650253851789302519620162063936706481729689040974288593097917383826669500135833989278 6","e":65537,"n":4488746876485737807536812361281060142441456247709423495749460998565243382425627992301822306649516903370266706154217696416353466583974323043040025907596830364583515921053226988093724172683135519194080606993327277051745418986916449541074016761}

Computed hash: 899328226

Decrypted signature: 899328226

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

THE JSON RESPONSE MESSAGE IS SHOWN HERE:

{"status":"success","message":"Total execution time to add this block was 108511 milliseconds\n"}

Number of Blocks on Chain == 6

We have a visitor

THE JSON REQUEST MESSAGE IS SHOWN HERE:

{"request":"add transaction","data":"Carol pays Donna 10 DSCoin","difficulty":4,"blockID":0,"clientID":"ba77d68adec2806639c519a16341e7621b0dac357be3e11169ef","signature":"619509344649138424533196414170813917013681767566175956931287742436286080521849220846959402096970154049097393012924514204870029943686489901712379236291833122779372324443498542042859942545313249202020633467639129196874354437255821661625123567","e":65537,"n":448874687648573780753681236128106014244145624770942349574946099856524338242562799230182230664951690337026670615421769641635346658397432304304002590759683036458351592105322698809372417268313551919408060699332727705174541898691644954107401676 1}

Computed hash: 668233196

Decrypted signature: 668233196

THE JSON RESPONSE MESSAGE IS SHOWN HERE:

{"status":"success","message":"Total execution time to add this block was 116002 milliseconds\n"}

Number of Blocks on Chain == 7

We have a visitor

THE JSON REQUEST MESSAGE IS SHOWN HERE:

{"request":"view","difficulty":0,"blockID":0,"clientID":"ba77d68adec2806639c519a16341e7621b0dac357be3e11169ef","signature":"100612469282391272199364 06

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

00570763608412118324341812768795256072723252041900455044603331836897295173002167177455323868628633252721881712324624150971623729932276216105734048145216237655291276912138793944247527444520930983361446847877199572404","e":65537,"n":448874687648573780753681236128106014244145624770942349574946099856524338242562799230182230664951690337026670615421769641635346658397432304304002590759683036458351592105322698809372417268313551919408060699332727705174541898691644954107401676 1}

Computed hash: 611021325

Decrypted signature: 611021325

THE JSON RESPONSE MESSAGE IS SHOWN HERE:

{"status":"success","message":"View the Blockchain\n{\n \"ds_chain\": [\n {\n \"index\": 0,\n \"time stamp \": \"2025-03-18 02:39:57.743\",\n \"Tx \": \"Genesis\",\n \"PrevHash\": \"\",\n \"nonce\": \"246\",\n \"difficulty\": 2\n },\n {\n \"index\": 1,\n \"time stamp \": \"2025-03-18 02:40:24.637\",\n \"Tx \": \"Alice pays Bob 100 DSCoin\",\n \"PrevHash\": \"00de2df1f886a84cb09e7f118e9772a19b29ac2e1666bdf132f2d86909b1e4a7\",\n \"nonce\": \"69462\",\n \"difficulty\": 4\n },\n {\n \"index\": 2,\n \"time stamp \": \"2025-03-18 02:40:33.492\",\n \"Tx \": \"Bob pays Tony 30 DSCoin\",\n \"PrevHash\": \"0000a6363742fe2ec6dde1e9c5adbbcbfc04c4fbffe15dbbedc83d93cfd6f7c7\",\n \"nonce\": \"50929\",\n \"difficulty\": 4\n },\n {\n \"index\": 3,\n \"time stamp \": \"2025-03-18 02:40:41.072\",\n \"Tx \": \"Carol pays Donna 10 DSCoin\",\n \"PrevHash\": \"000078b73644bccd2b8eabad72397900e6465628f1b3f7eeda0ac97d720e597c\",\n \"nonce\": \"36221\",\n \"difficulty\": 4\n },\n {\n \"index\": 4,\n \"time stamp \": \"2025-03-18 02:41:45.381\",\n \"Tx \": \"Alice pays Bob 100 DSCoin\",\n \"PrevHash\": \"f54354c322c1649a098978d8b9a63530ad6a1bb58adc74fe32403a0a5e76a127\",\n \"nonce\": \"108057\",\n \"difficulty\": 4\n },\n {\n

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

\"index\": 5,\n \"time stamp \": \"2025-03-18 02:41:53.142\",\n \"Tx \": \"Bob pays Carol 20 DSCoin\",\n \"PrevHash\": \"0000ed0c56e5a5addefc89f54d83bbc0151faf5884e43480e99f3d6369f561f0\",\n \"nonce\": \"31620\",\n \"difficulty\": 4\n },\n {\n \"index\": 6,\n \"time stamp \": \"2025-03-18 02:42:00.613\",\n \"Tx \": \"Carol pays Donna 10 DSCoin\",\n \"PrevHash\": \"000018785b95a3fa19347207fb85c58599bc2ee3ce7e90fbe771293dcf53490c\",\n \"nonce\": \"50034\",\n \"difficulty\": 4\n }\n],\n \"chainHash\": \"0000175c9f913eb1846c65afae7c8c02359cce218f403a0e0424182af341a299\"\n}\n"}

Number of Blocks on Chain == 7

We have a visitor

THE JSON REQUEST MESSAGE IS SHOWN HERE:

{"request":"verify","difficulty":0,"blockID":0,"clientID":"ba77d68adec2806639c519a16341e7621b0dac357be3e11169ef","signature":"40544380728139635361021538901581245649035680895365917143489617825019856992406559541632287065787170827096489475961179033408707121700553919395836431213171798883584402948365480913962806665284425077006475577467198066538488639764494090226979990 73","e":65537,"n":448874687648573780753681236128106014244145624770942349574946099856524338242562799230182230664951690337026670615421769641635346658397432304304002590759683036458351592105322698809372417268313551919408060699332727705174541898691644954107401676 1}

Computed hash: 4284235645

Decrypted signature: 4284235645

THE JSON RESPONSE MESSAGE IS SHOWN HERE:

{"status":"success","message":"Verifying entire chain\nChain verification: TRUE\nTotal execution time required to verify the chain was 125456 milliseconds\n"}

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

Number of Blocks on Chain == 7

We have a visitor

THE JSON REQUEST MESSAGE IS SHOWN HERE:

{"request":"corrupt","data":"Bob pays Tony 30 DSCoin","difficulty":0,"blockID":2,"clientID":"ba77d68adec2806639c519a16341e7621b0dac357be3e11169ef","signature":"26433412899686207207676288969300396763098993431829124732381252643822636817929712447174771147901959301919468161677614528167958355091911816470803066839520350557224600878052366677363008648971373451971280244235714267545090291758374578306328676888","e":65537,"n":448874687648573780753681236128106014244145624770942349574946099856524338242562799230182230664951690337026670615421769641635346658397432304304002590759683036458351592105322698809372417268313551919408060699332727705174541898691644954107401676 1}

Computed hash: 2625796698

Decrypted signature: 2625796698

THE JSON RESPONSE MESSAGE IS SHOWN HERE:

{"status":"success","message":"Block 2 now holds: Bob pays Tony 30 DSCoin\n"}

Number of Blocks on Chain == 7

We have a visitor

THE JSON REQUEST MESSAGE IS SHOWN HERE:

{"request":"view","difficulty":0,"blockID":0,"clientID":"ba77d68adec2806639c519a16341e7621b0dac357be3e11169ef","signature":"100612469282391272199364060057076360841211832434181276879525607272325204190045504460333183689729517300216717745532386862863325272188171232462415097162372993227621610573404814521623765529127691213879394424752744452093098336144684787719957240 4","e":65537,"n":448874687648573780753681236128106014244145624770942349574946099856524338242562799230182230664951690337026670615

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

421769641635346658397432304304002590759683036458351592105322698809372417268313551919408060699332727705174541898691644954107401676 1}

Computed hash: 611021325

Decrypted signature: 611021325

THE JSON RESPONSE MESSAGE IS SHOWN HERE:

{"status":"success","message":"View the Blockchain\n{\n \"ds_chain\": [\n {\n \"index\": 0,\n \"time stamp \": \"2025-03-18 02:39:57.743\",\n \"Tx \": \"Genesis\",\n \"PrevHash\": \"\",\n \"nonce\": \"246\",\n \"difficulty\": 2\n },\n {\n \"index\": 1,\n \"time stamp \": \"2025-03-18 02:40:24.637\",\n \"Tx \": \"Alice pays Bob 100 DSCoin\",\n \"PrevHash\": \"00de2df1f886a84cb09e7f118e9772a19b29ac2e1666bdf132f2d86909b1e4a7\",\n \"nonce\": \"69462\",\n \"difficulty\": 4\n },\n {\n \"index\": 2,\n \"time stamp \": \"2025-03-18 02:40:33.492\",\n \"Tx \": \"Bob pays Tony 30 DSCoin\",\n \"PrevHash\": \"0000a6363742fe2ec6dde1e9c5adbbcbfc04c4fbffe15dbbedc83d93cfd6f7c7\",\n \"nonce\": \"50929\",\n \"difficulty\": 4\n },\n {\n \"index\": 3,\n \"time stamp \": \"2025-03-18 02:40:41.072\",\n \"Tx \": \"Carol pays Donna 10 DSCoin\",\n \"PrevHash\": \"000078b73644bccd2b8eabad72397900e6465628f1b3f7eeda0ac97d720e597c\",\n \"nonce\": \"36221\",\n \"difficulty\": 4\n },\n {\n \"index\": 4,\n \"time stamp \": \"2025-03-18 02:41:45.381\",\n \"Tx \": \"Alice pays Bob 100 DSCoin\",\n \"PrevHash\": \"f54354c322c1649a098978d8b9a63530ad6a1bb58adc74fe32403a0a5e76a127\",\n \"nonce\": \"108057\",\n \"difficulty\": 4\n },\n {\n \"index\": 5,\n \"time stamp \": \"2025-03-18 02:41:53.142\",\n \"Tx \": \"Bob pays Carol 20 DSCoin\",\n \"PrevHash\": \"0000ed0c56e5a5addefc89f54d83bbc0151faf5884e43480e99f3d6369f561f0\",\n \"nonce\": \"31620\",\n \"difficulty\": 4\n },\n {\n \"index\": 6,\n \"time stamp \": \"2025-03-18 02:42:00.613\",\n \"Tx

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

\": \"Carol pays Donna 10 DSCoin\",\n \"PrevHash\": \"000018785b95a3fa19347207fb85c58599bc2ee3ce7e90fbe771293dcf53490c\",\n \"nonce\": \"50034\",\n \"difficulty\": 4\n }\n],\n \"chainHash\": \"0000175c9f913eb1846c65afae7c8c02359cce218f403a0e0424182af341a299\"\n}\n"}

Number of Blocks on Chain == 7

We have a visitor

THE JSON REQUEST MESSAGE IS SHOWN HERE:

{"request":"verify","difficulty":0,"blockID":0,"clientID":"ba77d68adec2806639c519a16341e7621b0dac357be3e11169ef","signature":"40544380728139635361021538901581245649035680895365917143489617825019856992406559541632287065787170827096489475961179033408707121700553919395836431213171798883584402948365480913962806665284425077006475577467198066538488639764494090226979990 73","e":65537,"n":44887468764857378075368123612810601424414562477094234957494609985652433824256279923018223066495169033702667061542176964163534665839743230430400259075968303645835159210532269880937241726831355191940806069933272770517454189869164495410740167 61}

Computed hash: 4284235645

Decrypted signature: 4284235645

THE JSON RESPONSE MESSAGE IS SHOWN HERE:

{"status":"success","message":"Verifying entire chain\nChain verification: TRUE\nTotal execution time required to verify the chain was 137804 milliseconds\n"}

Number of Blocks on Chain == 7

We have a visitor

THE JSON REQUEST MESSAGE IS SHOWN HERE:

{"request":"hide","difficulty":0,"blockID":0,"clientID":"ba77d68adec2806639c519a

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

16341e7621b0dac357be3e11169ef","signature":"319772286058196509792192171358141862952694393831604662410373436032928560631541711326526455862089499709700395905496728909581005794725245179265006367714206816324812951797569001274775122188587091112373716414637857011136485288436730048960966266","e":65537,"n":448874687648573780753681236128106014244145624770942349574946099856524338242562799230182230664951690337026670615421769641635346658397432304304002590759683036458351592105322698809372417268313551919408060699332727705174541898691644954107401676 1}

Computed hash: 1184219492

Decrypted signature: 1184219492

THE JSON RESPONSE MESSAGE IS SHOWN HERE:

{"status":"success","message":"Repairing the entire chain\nTotal execution time required to repair the chain was 139863 milliseconds\n"}

Number of Blocks on Chain == 7

We have a visitor

THE JSON REQUEST MESSAGE IS SHOWN HERE:

{"request":"verify","difficulty":0,"blockID":0,"clientID":"ba77d68adec2806639c519a16341e7621b0dac357be3e11169ef","signature":"405443807281396353610215389015812456490356808953659171434896178250198569924065595416322870657871708270964894759611790334087071217005539193958364312131717988835844029483654809139628066652844250770064755774671980665384886397644940902269799907 3","e":65537,"n":448874687648573780753681236128106014244145624770942349574946099856524338242562799230182230664951690337026670615421769641635346658397432304304002590759683036458351592105322698809372417268313551919408060699332727705174541898691644954107401676 1}

Computed hash: 4284235645

Decrypted signature: 4284235645

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

THE JSON RESPONSE MESSAGE IS SHOWN HERE:

{"status":"success","message":"Verifying entire chain\nChain verification: TRUE\nTotal execution time required to verify the chain was 144864 milliseconds\n"}

Number of Blocks on Chain == 7

We have a visitor

THE JSON REQUEST MESSAGE IS SHOWN HERE:

{"request":"view","difficulty":0,"blockID":0,"clientID":"ba77d68adec2806639c519a16341e7621b0dac357be3e11169ef","signature":"1006124692823912721993640600570763608412118324341812768795256072723252041900455044603331836897295173002167177455323868628633252721881712324624150971623729932276216105734048145216237655291276912138793944247527444520930983361446847877199572404","e":65537,"n":448874687648573780753681236128106014244145624770942349574946099856524338242562799230182230664951690337026670615421769641635346658397432304304002590759683036458351592105322698809372417268313551919408060699332727705174541898691644954107401676 1}

Computed hash: 611021325

Decrypted signature: 611021325

THE JSON RESPONSE MESSAGE IS SHOWN HERE:

{"status":"success","message":"View the Blockchain\n{\n \"ds_chain\": [\n {\n \"index\": 0,\n \"time stamp \": \"2025-03-18 02:39:57.743\",\n \"Tx \": \"Genesis\",\n \"PrevHash\": \"\",\n \"nonce\": \"246\",\n \"difficulty\": 2\n },\n {\n \"index\": 1,\n \"time stamp \": \"2025-03-18 02:40:24.637\",\n \"Tx \": \"Alice pays Bob 100 DSCoin\",\n \"PrevHash\": \"00de2df1f886a84cb09e7f118e9772a19b29ac2e1666bdf132f2d86909b1e4a7\",\n \"nonce\": \"69462\",\n \"difficulty\": 4\n },\n {\n \"index\": 2,\n \"time stamp \": \"2025-03-18 02:40:33.492\",\n \"Tx

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

**\": \"Bob pays Tony 30 DSCoin\",\n \"PrevHash\": \"0000a6363742fe2ec6dde1e9c5adbbcbfc04c4fbffe15dbbedc83d93cfd6f7c7\",\n \"nonce\": \"50929\",\n \"difficulty\": 4\n },\n {\n \"index\": 3,\n \"time stamp \": \"2025-03-18 02:40:41.072\",\n \"Tx \": \"Carol pays Donna 10 DSCoin\",\n \"PrevHash\": \"000078b73644bccd2b8eabad72397900e6465628f1b3f7eeda0ac97d720e597c\",\n \"nonce\": \"121961\",\n \"difficulty\": 4\n },\n {\n \"index\": 4,\n \"time stamp \": \"2025-03-18 02:41:45.381\",\n \"Tx \": \"Alice pays Bob 100 DSCoin\",\n \"PrevHash\": \"000092ca2831aa8be048b1678a3d3b740fab93e08fb63d42318e84e6f917edd1\",\n \"nonce\": \"108057\",\n \"difficulty\": 4\n },\n {\n \"index\": 5,\n \"time stamp \": \"2025-03-18 02:41:53.142\",\n \"Tx \": \"Bob pays Carol 20 DSCoin\",\n \"PrevHash\": \"1a8311788a5564cf2fab754c163df704d59d9cf556a49ac4e45d9d53ba83d950\",\n \"nonce\": \"31620\",\n \"difficulty\": 4\n },\n {\n \"index\": 6,\n \"time stamp \": \"2025-03-18 02:42:00.613\",\n \"Tx \": \"Carol pays Donna 10 DSCoin\",\n \"PrevHash\": \"9aa5c8f3f9230fa199c105165ec725420d09d72823cbd123b89100fa1134977c\",\n \"nonce\": \"50034\",\n \"difficulty\": 4\n }\n],\n \"chainHash\": \"5eca3b5f9fc42bb62ec2ffc3724c8ccbbe887979a2c955ef148796ff8cfeb250\"\n}\n"
}

Number of Blocks on Chain == 7

We have a visitor

THE JSON REQUEST MESSAGE IS SHOWN HERE:

{"request":"status","difficulty":0,"blockID":0,"clientID":"eece460e9777637970086 1e305e990dfc6290f7b43fa962eead4","signature":"4543267095302432051778528800508713779749491025880902929893761624700374454690317587306051234678088464462500604587831990589836216166025777326314223671423302662003010541581484789120369109155728460299734905588654636693217629591752886892560877 18","e":65537,"n":19136418018161755523484543388802982791134702

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

086739511890132536342114186313452098139359468136879166756834957191800917771071500242482549938559369067714311288756960880437650473553701255061997132470845388004505627486121490104153437452184403056635}

Computed hash: 2995273089

Decrypted signature: 2995273089

THE JSON RESPONSE MESSAGE IS SHOWN HERE:

{"status":"success","message":"Current size of chain: 7\nDifficulty of most recent block: 4\nTotal difficulty for all blocks: 26\nApproximate hashes per second: 0\nExpected total hashes required: 393472.0\nNonce for most recent block: 50034\nChain hash: 5eca3b5f9fc42bb62ec2ffc3724c8ccbbe887979a2c955ef148796ff8cfeb250\n"}

Number of Blocks on Chain == 7

We have a visitor

THE JSON REQUEST MESSAGE IS SHOWN HERE:

{"request":"add transaction","data":"Alice pays Bob 100 DSCoin","difficulty":4,"blockID":0,"clientID":"eece460e97776379700861e305e990dfc6290f7b43fa962eead4","signature":"1149021273782741613077444946530275169149247424822760825758438570613772855811051338068456675928876790771465975953033806813939672927834298273911948441541782720028095021162706712945073559615972532046875542809688260949309419892735314125914709340","e":65537,"n":191364180181617555234845433888029827911347020867395118901325363421141863134520981393594681368791667568349571918009177107150024248254993855936906771431128875696088043765047355370125506199713247084538800450562748612149010415343745218440305663}

Computed hash: 209940527

Decrypted signature: 209940527

THE JSON RESPONSE MESSAGE IS SHOWN HERE:

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

{"status":"success","message":"Total execution time to add this block was 165517 milliseconds\n"}

Number of Blocks on Chain == 8

We have a visitor

THE JSON REQUEST MESSAGE IS SHOWN HERE:

{"request":"add transaction","data":"Bob pays Carol 20 DSCoin","difficulty":4,"blockID":0,"clientID":"eece460e97776379700861e305e990dfc6290f7b43fa962eead4","signature":"13471576614357660973240336196221701312126198374392672296395854945539430735825888581911718321788246286118878028165117256422314341594347452781670196715376627202698199369938630874582919617211222248544583241490211553628251954944003746441153582 62","e":65537,"n":19136418018161755523484543388802982791134702086739511890132536342114186313452098139359468136879166756834957191800917771071500242482549938559369067714311288756960880437650473553701255061997132470845388004505627486121490104153437452184403056 63}

Computed hash: 499854069

Decrypted signature: 499854069

THE JSON RESPONSE MESSAGE IS SHOWN HERE:

{"status":"success","message":"Total execution time to add this block was 175728 milliseconds\n"}

Number of Blocks on Chain == 9

We have a visitor

THE JSON REQUEST MESSAGE IS SHOWN HERE:

{"request":"add transaction","data":"Carol pays Donna 10 DSCoin","difficulty":4,"blockID":0,"clientID":"eece460e97776379700861e305e990dfc6290f7b43fa962eead4","signature":"18227379303018665160558992722987170509039776250253742556544835452853702456457855368273786846688991896 71

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

974195842683222402046981405905436528538823269026319004349429992099513637962891067621419262537567331699084457266339926171138787911831441607","e":65537,"n":191364180181617555234845433888029827911347020867395118901325363421141863134520981393594681368791667568349571918009177710715002424825499385593690677143112887569608804376504735537012550619971324708453880045056274861214901041534374521844030566 3}

Computed hash: 2567887451

Decrypted signature: 2567887451

THE JSON RESPONSE MESSAGE IS SHOWN HERE:

{"status":"success","message":"Total execution time to add this block was 181314 milliseconds\n"}

Number of Blocks on Chain == 10

We have a visitor

THE JSON REQUEST MESSAGE IS SHOWN HERE:

{"request":"view","difficulty":0,"blockID":0,"clientID":"eece460e97776379700861e305e990dfc6290f7b43fa962eead4","signature":"189929307541776555105565110570924001270251310255100304304205640681794646694716969316296759353977414513330105123908838005800057401021050742631653381129167253959575989271074203889658912241256323186944179655035727850301233624674786191026212332 6","e":65537,"n":191364180181617555234845433888029827911347020867395118901325363421141863134520981393594681368791667568349571918009177107150024248254993855936906771431128875696088043765047355370125506199713247084538800450562748612149010415343745218440305663}

Computed hash: 382328642

Decrypted signature: 382328642

THE JSON RESPONSE MESSAGE IS SHOWN HERE:

{"status":"success","message":"View the Blockchain\n{\n \"ds_chain\": [\n

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

{\n \"index\": 0,\n \"time stamp \": \"2025-03-18 02:39:57.743\",\n \"Tx \": \"Genesis\",\n \"PrevHash\": \"\",\n \"nonce\": \"246\",\n \"difficulty\": 2\n },\n {\n \"index\": 1,\n \"time stamp \": \"2025-03-18 02:40:24.637\",\n \"Tx \": \"Alice pays Bob 100 DSCoin\",\n \"PrevHash\": \"00de2df1f886a84cb09e7f118e9772a19b29ac2e1666bdf132f2d86909b1e4a7\",\n \"nonce\": \"69462\",\n \"difficulty\": 4\n },\n {\n \"index\": 2,\n \"time stamp \": \"2025-03-18 02:40:33.492\",\n \"Tx \": \"Bob pays Tony 30 DSCoin\",\n \"PrevHash\": \"0000a6363742fe2ec6dde1e9c5adbbcbfc04c4fbffe15dbbedc83d93cfd6f7c7\",\n \"nonce\": \"50929\",\n \"difficulty\": 4\n },\n {\n \"index\": 3,\n \"time stamp \": \"2025-03-18 02:40:41.072\",\n \"Tx \": \"Carol pays Donna 10 DSCoin\",\n \"PrevHash\": \"000078b73644bccd2b8eabad72397900e6465628f1b3f7eeda0ac97d720e597c\",\n \"nonce\": \"121961\",\n \"difficulty\": 4\n },\n {\n \"index\": 4,\n \"time stamp \": \"2025-03-18 02:41:45.381\",\n \"Tx \": \"Alice pays Bob 100 DSCoin\",\n \"PrevHash\": \"000092ca2831aa8be048b1678a3d3b740fab93e08fb63d42318e84e6f917edd1\",\n \"nonce\": \"108057\",\n \"difficulty\": 4\n },\n {\n \"index\": 5,\n \"time stamp \": \"2025-03-18 02:41:53.142\",\n \"Tx \": \"Bob pays Carol 20 DSCoin\",\n \"PrevHash\": \"1a8311788a5564cf2fab754c163df704d59d9cf556a49ac4e45d9d53ba83d950\",\n \"nonce\": \"31620\",\n \"difficulty\": 4\n },\n {\n \"index\": 6,\n \"time stamp \": \"2025-03-18 02:42:00.613\",\n \"Tx \": \"Carol pays Donna 10 DSCoin\",\n \"PrevHash\": \"9aa5c8f3f9230fa199c105165ec725420d09d72823cbd123b89100fa1134977c\",\n \"nonce\": \"50034\",\n \"difficulty\": 4\n },\n {\n \"index\": 7,\n \"time stamp \": \"2025-03-18 02:42:48.369\",\n \"Tx \": \"Alice pays Bob 100 DSCoin\",\n \"PrevHash\": \"5eca3b5f9fc42bb62ec2ffc3724c8ccbbe887979a2c955ef148796ff8cfeb250\",\n \"nonce\": \"46870\",\n \"difficulty\": 4\n },\n {\n

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

\"index\": 8,\n \"time stamp \": \"2025-03-18 02:42:58.63\",\n \"Tx \": \"Bob pays Carol 20 DSCoin\",\n \"PrevHash\": \"000056d586208a739068dbcbf457feb9ee909e5586d9c5bb04244e50d12a2aa1\",\n \"nonce\": \"16300\",\n \"difficulty\": 4\n },\n {\n \"index\": 9,\n \"time stamp \": \"2025-03-18 02:43:04.202\",\n \"Tx \": \"Carol pays Donna 10 DSCoin\",\n \"PrevHash\": \"00005765b1bafba6721013b44ba234b8356efaf31c89d5c76d76a019983ffd11\",\n \"nonce\": \"25438\",\n \"difficulty\": 4\n }\n],\n \"chainHash\": \"0000fea1d82e87157a0096546801034429e4c1de50b177364831ffc8706743c2\"\n }\n"}

Number of Blocks on Chain == 10

We have a visitor

THE JSON REQUEST MESSAGE IS SHOWN HERE:

{"request":"verify","difficulty":0,"blockID":0,"clientID":"eece460e97776379700861e305e990dfc6290f7b43fa962eead4","signature":"184574394548705401035465542470868194396360792775369371497006764740227804098133241131865028508076525572963645403827982081340937006296284865417016183260528116993191010518016354877047385803196349447223943298702298100940603899593074827026049028 1","e":65537,"n":191364180181617555234845433888029827911347020867395118901325363421141863134520981393594681368791667568349571918009177710715002424825499385593690677143112887569608804376504735537012550619971324708453880045056274861214901041534374521844030566 3}

Computed hash: 4084478290

Decrypted signature: 4084478290

THE JSON RESPONSE MESSAGE IS SHOWN HERE:

{"status":"success","message":"Verifying entire chain\nChain verification: TRUE\nTotal execution time required to verify the chain was 187972 milliseconds\n"}

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

Number of Blocks on Chain == 10

We have a visitor

THE JSON REQUEST MESSAGE IS SHOWN HERE:

{"request":"corrupt","data":"Bob pays Tony 30 DSCoin","difficulty":0,"blockID":2,"clientID":"eece460e97776379700861e305e990dfc6290f7b43fa962eead4","signature":"8209154068330111368601439494794587790245529333456810507021340197884099417395463238213883278242523520487853027729514625065866012967379400278610156053610396661099011157355697758225594248900229467994276989408813776833048398124298277845465648074","e":65537,"n":19136418018161755523484543388802982791134702086739511890132536342114186313452098139359468136879166756834957191800917771071500242482549938559369067714311288756960880437650473553701255061997132470845388004505627486121490104153437452184403056634}

Computed hash: 2133429677

Decrypted signature: 2133429677

THE JSON RESPONSE MESSAGE IS SHOWN HERE:

{"status":"success","message":"Block 2 now holds: Bob pays Tony 30 DSCoin\n"}

Number of Blocks on Chain == 10

We have a visitor

THE JSON REQUEST MESSAGE IS SHOWN HERE:

{"request":"view","difficulty":0,"blockID":0,"clientID":"eece460e97776379700861e305e990dfc6290f7b43fa962eead4","signature":"1899293075417765551055651105709240012702513102551003043042056406817946466947169693162967593539774145133301051239088380058000574010210507426316533811291672539595759892710742038896589122412563231869441796550357278503012336246747861910262123326","e":65537,"n":19136418018161755523484543388802982791134702086739511890132536342114186313452098139359468136879166756834957191800

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

0917771071500242482549938559369067714311288756960880437650473553701255061997132470845388004505627486121490104153437452184403056 63}

Computed hash: 382328642

Decrypted signature: 382328642

THE JSON RESPONSE MESSAGE IS SHOWN HERE:

{"status":"success","message":"View the Blockchain\n{\n \"ds_chain\": [\n {\n \"index\": 0,\n \"time stamp \": \"2025-03-18 02:39:57.743\",\n \"Tx \": \"Genesis\",\n \"PrevHash\": \"\",\n \"nonce\": \"246\",\n \"difficulty\": 2\n },\n {\n \"index\": 1,\n \"time stamp \": \"2025-03-18 02:40:24.637\",\n \"Tx \": \"Alice pays Bob 100 DSCoin\",\n \"PrevHash\": \"00de2df1f886a84cb09e7f118e9772a19b29ac2e1666bdf132f2d86909b1e4a7\",\n \"nonce\": \"69462\",\n \"difficulty\": 4\n },\n {\n \"index\": 2,\n \"time stamp \": \"2025-03-18 02:40:33.492\",\n \"Tx \": \"Bob pays Tony 30 DSCoin\",\n \"PrevHash\": \"0000a6363742fe2ec6dde1e9c5adbbcbfc04c4fbffe15dbbedc83d93cfd6f7c7\",\n \"nonce\": \"50929\",\n \"difficulty\": 4\n },\n {\n \"index\": 3,\n \"time stamp \": \"2025-03-18 02:40:41.072\",\n \"Tx \": \"Carol pays Donna 10 DSCoin\",\n \"PrevHash\": \"000078b73644bccd2b8eabad72397900e6465628f1b3f7eeda0ac97d720e597c\",\n \"nonce\": \"121961\",\n \"difficulty\": 4\n },\n {\n \"index\": 4,\n \"time stamp \": \"2025-03-18 02:41:45.381\",\n \"Tx \": \"Alice pays Bob 100 DSCoin\",\n \"PrevHash\": \"000092ca2831aa8be048b1678a3d3b740fab93e08fb63d42318e84e6f917edd1\",\n \"nonce\": \"108057\",\n \"difficulty\": 4\n },\n {\n \"index\": 5,\n \"time stamp \": \"2025-03-18 02:41:53.142\",\n \"Tx \": \"Bob pays Carol 20 DSCoin\",\n \"PrevHash\": \"1a8311788a5564cf2fab754c163df704d59d9cf556a49ac4e45d9d53ba83d950\",\n \"nonce\": \"31620\",\n \"difficulty\": 4\n },\n {\n \"index\": 6,\n \"time stamp \": \"2025-03-18 02:42:00.613\",\n \"Tx

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

\": \"Carol pays Donna 10 DSCoin\",\n \"PrevHash\": \"9aa5c8f3f9230fa199c105165ec725420d09d72823cbd123b89100fa1134977c\",\n \"nonce\": \"50034\",\n \"difficulty\": 4\n },\n {\n \"index\": 7,\n \"time stamp \": \"2025-03-18 02:42:48.369\",\n \"Tx \": \"Alice pays Bob 100 DSCoin\",\n \"PrevHash\": \"5eca3b5f9fc42bb62ec2ffc3724c8ccbbe887979a2c955ef148796ff8cfeb250\",\n \"nonce\": \"46870\",\n \"difficulty\": 4\n },\n {\n \"index\": 8,\n \"time stamp \": \"2025-03-18 02:42:58.63\",\n \"Tx \": \"Bob pays Carol 20 DSCoin\",\n \"PrevHash\": \"000056d586208a739068dbcbf457feb9ee909e5586d9c5bb04244e50d12a2aa1\",\n \"nonce\": \"16300\",\n \"difficulty\": 4\n },\n {\n \"index\": 9,\n \"time stamp \": \"2025-03-18 02:43:04.202\",\n \"Tx \": \"Carol pays Donna 10 DSCoin\",\n \"PrevHash\": \"00005765b1bafba6721013b44ba234b8356efaf31c89d5c76d76a019983ffd11\",\n \"nonce\": \"25438\",\n \"difficulty\": 4\n }\n],\n \"chainHash\": \"0000fea1d82e87157a0096546801034429e4c1de50b177364831ffc8706743c2\"\n }\n"}

Number of Blocks on Chain == 10

We have a visitor

THE JSON REQUEST MESSAGE IS SHOWN HERE:

{"request":"verify","difficulty":0,"blockID":0,"clientID":"eece460e97776379700861e305e990dfc6290f7b43fa962eead4","signature":"184574394548705401035465542470868194396360792775369371497006764740227804098133241131865028508076525572963645403827982081340937006296284865417016183260528116993191010518016354877047385803196349447223943298702298100940603899593074827026049028 1","e":65537,"n":191364180181617555234845433888029827911347020867395118901325363421141863134520981393594681368791667568349571918009177710715002424825499385593690677143112887569608804376504735537012550619971324708453880045056274861214901041534374521844030566 3}

Computed hash: 4084478290

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

Decrypted signature: 4084478290

THE JSON RESPONSE MESSAGE IS SHOWN HERE:

{"status":"success","message":"Verifying entire chain\nChain verification: TRUE\nTotal execution time required to verify the chain was 199971 milliseconds\n"}

Number of Blocks on Chain == 10

We have a visitor

THE JSON REQUEST MESSAGE IS SHOWN HERE:

{"request":"hide","difficulty":0,"blockID":0,"clientID":"eece460e97776379700861e305e990dfc6290f7b43fa962eead4","signature":"290509127954117379685723765852352080812457342811196939333418269507935081255075763650246413690770341798349977903379095625330339296029715332795765990172685923570625756963068250407393760289302867759178305395923955530693059004774997671378870108","e":65537,"n":191364180181617555234845433888029827911347020867395118901325363421141863134520981393594681368791667568349571918009177710715002424825499385593690677143112887569608804376504735537012550619971324708453880045056274861214901041534374521844030566 3}

Computed hash: 1973667155

Decrypted signature: 1973667155

THE JSON RESPONSE MESSAGE IS SHOWN HERE:

{"status":"success","message":"Repairing the entire chain\nTotal execution time required to repair the chain was 209006 milliseconds\n"}

Number of Blocks on Chain == 10

We have a visitor

THE JSON REQUEST MESSAGE IS SHOWN HERE:

{"request":"verify","difficulty":0,"blockID":0,"clientID":"eece460e97776379700861

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

**e305e990dfc6290f7b43fa962eead4","signature":"184574394548705401035465542
470868194396360792775369371497006764740227804098133241131865028508076525572963645403827982081340937006296284865417016183260528116993191010518016354877047385803196349447223943298702298100940603899593074827026049028 1","e":65537,"n":191364180181617555234845433888029827911347020867395118901325363421141863134520981393594681368791667568349571918009177710715002424825499385593690677143112887569608804376504735537012550619971324708453880045056274861214901041534374521844030566 3}**

Computed hash: 4084478290

Decrypted signature: 4084478290

THE JSON RESPONSE MESSAGE IS SHOWN HERE:

{"status":"success","message":"Verifying entire chain\nChain verification: TRUE\nTotal execution time required to verify the chain was 215813 milliseconds\n"}

Number of Blocks on Chain == 10

We have a visitor

THE JSON REQUEST MESSAGE IS SHOWN HERE:

{"request":"view","difficulty":0,"blockID":0,"clientID":"eece460e97776379700861e305e990dfc6290f7b43fa962eead4","signature":"189929307541776555105565110570924001270251310255100304304205640681794646694716969316296759353977414513330105123908838005800057401021050742631653381129167253959575989271074203889658912241256323186944179655035727850301233624674786191026212332 6","e":65537,"n":191364180181617555234845433888029827911347020867395118901325363421141863134520981393594681368791667568349571918009177710715002424825499385593690677143112887569608804376504735537012550619971324708453880045056274861214901041534374521844030566 3}

Computed hash: 382328642

Decrypted signature: 382328642

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

THE JSON RESPONSE MESSAGE IS SHOWN HERE:

{"status":"success","message":"View the Blockchain\n{\n \"ds_chain\": [\n {\n \"index\": 0,\n \"time stamp \": \"2025-03-18 02:39:57.743\",\n \"Tx \": \"Genesis\",\n \"PrevHash\": \"\",\n \"nonce\": \"246\",\n \"difficulty\": 2\n },\n {\n \"index\": 1,\n \"time stamp \": \"2025-03-18 02:40:24.637\",\n \"Tx \": \"Alice pays Bob 100 DSCoin\",\n \"PrevHash\": \"00de2df1f886a84cb09e7f118e9772a19b29ac2e1666bdf132f2d86909b1e4a7\",\n \"nonce\": \"69462\",\n \"difficulty\": 4\n },\n {\n \"index\": 2,\n \"time stamp \": \"2025-03-18 02:40:33.492\",\n \"Tx \": \"Bob pays Tony 30 DSCoin\",\n \"PrevHash\": \"0000a6363742fe2ec6dde1e9c5adbbcbfc04c4fbffe15dbbedc83d93cfd6f7c7\",\n \"nonce\": \"50929\",\n \"difficulty\": 4\n },\n {\n \"index\": 3,\n \"time stamp \": \"2025-03-18 02:40:41.072\",\n \"Tx \": \"Carol pays Donna 10 DSCoin\",\n \"PrevHash\": \"000078b73644bccd2b8eabad72397900e6465628f1b3f7eeda0ac97d720e597c\",\n \"nonce\": \"121961\",\n \"difficulty\": 4\n },\n {\n \"index\": 4,\n \"time stamp \": \"2025-03-18 02:41:45.381\",\n \"Tx \": \"Alice pays Bob 100 DSCoin\",\n \"PrevHash\": \"000092ca2831aa8be048b1678a3d3b740fab93e08fb63d42318e84e6f917edd1\",\n \"nonce\": \"194450\",\n \"difficulty\": 4\n },\n {\n \"index\": 5,\n \"time stamp \": \"2025-03-18 02:41:53.142\",\n \"Tx \": \"Bob pays Carol 20 DSCoin\",\n \"PrevHash\": \"00000b2b67dae3ecba5cfa5427b79ee47f85c362bede1f3a128fd2490c5aa069\",\n \"nonce\": \"249128\",\n \"difficulty\": 4\n },\n {\n \"index\": 6,\n \"time stamp \": \"2025-03-18 02:42:00.613\",\n \"Tx \": \"Carol pays Donna 10 DSCoin\",\n \"PrevHash\": \"c5eb2ba7292578fa471e9f7c92c8de5d53c112d81aa6627e0552075bd2cbd422\",\n \"nonce\": \"89065\",\n \"difficulty\": 4\n },\n {\n \"index\": 7,\n \"time stamp \": \"2025-03-18 02:42:48.369\",\n \"Tx \": \"Alice pays Bob 100 DSCoin\",\n \"PrevHash\":

Course: Distribution System Management
Instructor: Prof. McCarthy, Prof. Barrett
Name: Jerry Huang (Tzu-Chieh Huang)
Andrew ID: jerryh

\"72e7901d3ede52a3bc31be2b13536e2d311c2e691576de78b66662c544e5d1dd\",\n \"nonce\": \"46870\",\n \"difficulty\": 4\n },\n {\n \"index\": 8,\n \"time stamp \": \"2025-03-18 02:42:58.63\",\n \"Tx \": \"Bob pays Carol 20 DSCoin\",\n \"PrevHash\": \"1794a86aa4f794eba0baf1ae6b80eb607a83c97645f09357763223513162d9d3\",\n \"nonce\": \"16300\",\n \"difficulty\": 4\n },\n {\n \"index\": 9,\n \"time stamp \": \"2025-03-18 02:43:04.202\",\n \"Tx \": \"Carol pays Donna 10 DSCoin\",\n \"PrevHash\": \"43c2f6654a3e83e7bdb00f4582dec6ccecf222033379d30269e96f4d4523ec72\",\n \"nonce\": \"25438\",\n \"difficulty\": 4\n }\n],\n \"chainHash\": \"7e0394a2848727eb8f665fdc7e7b8a13398b2345ffdb9d993ebf078972047fc5\"\n}\n\"}

Number of Blocks on Chain == 10