

## CSI 2510 – Assignment 2

### 1- Classes and Methods

\*\*\*Please note that the classes are very well commented if further explanation is needed\*\*\*

- **Transaction:** The Transaction class represents the transaction between a sender and a receiver in one Block of a Blockchain. There is also an amount associated with this Transaction called amount. Therefore, three class variables are needed: sender, receiver and amount.
  - **Constructor:** takes the 3 variables as parameters and initializes them
  - **Methods:** getters for each variable that return the variable itself (getSender(), getReceiver() and getAmount()). Also, a toString() method to get the string version of the object Transaction.
- **Block:** The Block class stores one transaction as well as a timestamp, nonce, previous hash, current hash and the index in the blockchain. Therefore, the class variables needed are: index, timestamp, transaction, nonce, previousHash and hash.
  - **Constructor:** 2 constructors are needed in this class, one is used when reading from a file and creating blocks and the other when creating new transactions and hence a new block for that transaction.
    - **Block(index, timestamp, transaction, nonce, previousHash, hash)**
    - **Block(index, timestamp, transaction, previousHash)**
  - **Methods:** many getters, few setters and one toString method were used. I created getters for all class variables (getIndex(), getTimestamp(), getNonce(), getPreviousHash(), getHash()). The setters were only needed for the missing parameters from the second constructor because once they are calculated (nonce and hash), they can be set in the Block class (setNonce(String nonce), setHash(String hash)). Finally, a toString() method was used to create the string that would produce the hash code.
- **Blockchain:** The Blockchain class represents the blockchain storing all the blocks which each have one transaction. The class variable is therefore an arraylist called blocks
  - **Constructor:** There is only one constructor which initializes the arraylist (blockchain)
  - **Methods:**
    - public static fromFile(String fileName) - reads blockchain from text file
    - public void toFile (String filename) - outputs blockchain to text file
    - public Boolean validateBlockchain() – validates blockchain (returns true if valid and false otherwise)
    - public int getBalance(String username) – returns the balance of the user depending on other transactions in the blockchain
    - public void add(Block block) – adds block to the blockchain

- `public String generateHash(Block block)` – generates hash code based on the toString() method of the given block.
- `public void generateNonce(Block block)` – generates nonce to fulfill hash condition (start with 5 0s).
- `public static void main(String[] args)` – main program

○ **Proof of Work Algorithm:**

*While the nonce producing the correct hash is not found, do:*

*Generate random integer of length 3-7 (nonceLength)*

*Initialize character array of length nonceLength*

*for index 0 to (nonceLength -1) do:*

*generate random integers from 33-126 inclusively*

*populate character array with casted random integers*

*Create string from this randomized character array*

*Set the nonce of the block to that randomized character array*

*Find hash corresponding to nonce and set the nonce to the block*

*If condition of 5 0s is met for hash code*

*end while loop, if not, continue looping.*

## 2- Summary Table

The following table is a summary of the transactions that were created and found in the **blockchain\_ehagg014.txt** text file:

Created Transaction	Number of Hash Trials
1	561803
2	1262972
3	181080
4	1049059
5	48093
6	106739
7	2763659
8	1241874
9	1857119
10	829729
<b>Average Number Of Trials</b>	990212.7

### 3- Additional Information

- I have validated 2 miners: vhowa026 and osiag057 since both had asked me to validate their blockchains. These two valid blockchains can be found in the following text files: **blockchain\_osiag057\_ehagg014.txt** and **blockchain\_vhowa026\_ehagg014.txt**
- My blockchain (**blockchain\_ehagg014.txt**) was validated by three miners: bjeau025, dngen049 and vhowa026 in the following text files: **blockchain\_ehagg014\_bjeau025.txt**, **blockchain\_ehagg014\_dngen049.txt** and **blockchain\_ehagg014\_vhowa026.txt**