National University of Computer and Emerging Sciences

Blockchain and Cryptocurrency (CS4049)

Date: February 27th 2024

Course Instructor(s)

Syeda Tayyaba Bukhari

Sessional-I Exam

Total Time: 1 Hours
Total Marks: 25
Total Questions: 03

Semester: Spring 2024
Campus: Lahore
Dept: Computer Science

Student Name	Roll No	Section	Student Signature
_____	_____	_____	_____

Vetted by _____ Vetter Signature _____

Instructions:

1. Make sure there are total 6 pages including title page.
2. All questions are to be attempted on this paper. **No extra Sheets are allowed**
3. Understanding of question is the part of exam.
4. If there is any ambiguity in the paper, benefit will be given to students.

Question No.	1	2	3	Total
Total Marks	6	10	9	25
Obtained Marks				

DO NOT OPEN UNTIL YOU ARE TOLD TO DO SO…..GOOD LUCK 😊

Question 1: Choose the Best Answer. Write your choice in above table either A, B, C or D
Answer Section for Q1 (Any type of overwriting is not allowed): [6 marks]

1	B
2	C
3	D
4	C
5	C
6	A

1. An orphan block is only created when 51% attack is successful
   A. True
   B. False
2. If two miners, A and B, solve puzzle at a same time whose block will be the part of blockchain?
   A. The block that reaches the majority of nodes first will win
   B. The block that was broadcast first will win
   C. The miner who finds the next block will likely resolve the tie
   D. Each node has its own version of the blockchain
3. What is the purpose of nonce in the block chain?
   A. A block identity
   B. Gives identity to miners
   C. It makes mining easy
   D. None of above
4. What is the main benefit of blockchain technology?
   A. Increased transaction speed
   B. Centralized control over data
   C. Improved data security and integrity
   D. Lower energy consumption
5. What does immutability mean in the context of blockchain?
   A. The ability to modify data on the blockchain
   B. The capability to upgrade the blockchain protocol
   C. The permanence and inability to alter recorded data
   D. The feature that allows for anonymous transactions
6. What is the role of a miner in a proof-of-work blockchain network?
   A. Verifying transactions and adding them to the blockchain
   B. Creating new cryptocurrencies
   C. Facilitating peer-to-peer transactions
   D. Establishing consensus among network participants

Q2: Answer following questions:

a. Describe a potential use case of blockchain in the music industry. [2 marks]

Blockchain can be utilized in the music industry to address copyright issues, track ownership, and ensure fair compensation for artists. Additionally, blockchain can create a transparent and immutable record of ownership for music rights, simplifying licensing processes and reducing disputes over intellectual property.

b. What are two main challenges (Only names) of Consensus Protocol? [2 marks]

c. What do you know about Byzantine Fault Tolerance? [2 marks]

d. Nonce stands for? What is Golden Nonce? **[1+1 marks]**

e. How mining works after adding timestamp attribute in Block? **[2 marks]**

Q3: Complete following incomplete functions (highlighted in bold) in GoLang [4+5 marks]

Part 1:

```
type Block struct {
        Spender     map[string]int //Spender is an array of integers in which the indexes are strings
        Receiver    map[string]int
        PrevPointer *Block
        PrevHash    string
        CurrentHash string
}
func CalculateBalance(userName string, chainHead *Block) int {
//calculate balance of a specific user

        var balance = 0
        var amountSpend = 0
        var amountReceived = 0
        var tempBlock = chainHead
        for tempBlock != nil {
                amountReceived += tempBlock.Receiver[userName]
                amountSpend += tempBlock.Spender[userName]
                tempBlock = tempBlock.PrevPointer
        }
        balance = amountReceived - amountSpend
        return balance

}
```

Part 2:

```
type Block struct {
        transactions []string
        prevPointer  *Block
        prevHash     string
        currentHash  string
}
func CalculateHash(inputBlock *Block) string {
```

```
	hash := sha256.Sum256([]byte(fmt.Sprintln(inputBlock)))
	return hex.EncodeToString(hash[:])
}
func InsertBlock(transactionsToInsert []string, chainHead *Block) *Block {
//inserting new Block
	if chainHead == nil {
		chainHead = &Block{} //creating new block
		chainHead.transactions = transactionsToInsert
		chainHead.prevPointer=nil
		chainHead.prevHash=""
		chainHead.currentHash = CalculateHash(chainHead)
		return chainHead









	else{

		var newBlock *Block
		newBlock = &Block{}

		newBlock.transactions = transactionsToInsert
		newBlock.prevPointer = chainHead
		newBlock.prevHash = CalculateHash(chainHead)
		newBlock.currentHash = CalculateHash(newBlock)

		chainHead = newBlock

		return chainHead









	}

}
```