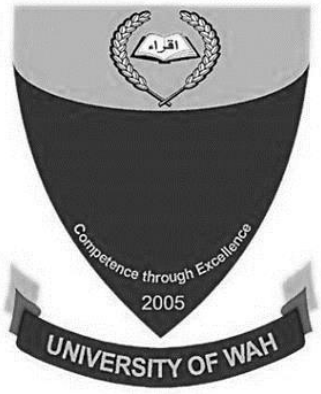
BlockChain Technology

Assignment 2



Submitted by
Khurram Zaman

UW-21-CS-BS-091

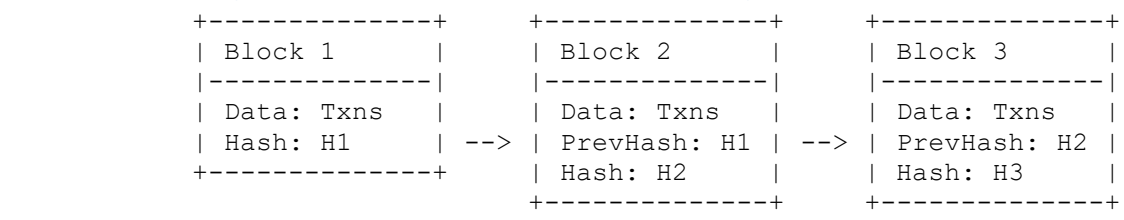
Submitted to
Ms Salma Bukhari

Department of Computer Science

University of Wah

Q1. (a) Apply the concept of immutability by illustrating how data is stored and linked in a blockchain. [10 Marks]

Diagram: Blockchain Block Structure and Linking



Immutability Works in Blockchain

1. **Block Hashing:** Each block in a blockchain contains:
    - **Transaction data**
    - **A cryptographic hash** of the current block's data
    - **The hash of the previous block** (linking them together)
2. **Immutability:**
    - If someone tries to **alter data** in Block 1, its hash (`H1`) will change.
    - This causes a mismatch in Block 2's `PrevHash` (which still holds old `H1`), thereby invalidating all subsequent blocks.
    - Fixing this mismatch would require recalculating all subsequent hashes **in real time**, which is computationally infeasible due to the **Proof-of-Work** or **consensus mechanisms** in place.
3. **Why Immutability Ensures Secure Record-Keeping:**
    - Prevents **unauthorized tampering**.
    - Creates a **trustless environment**, where records don't require third-party verification.
    - Provides a **verifiable audit trail** for sensitive data such as financial records, supply chain data, or identity records.

Q1. (b) Implement access control in a permissioned blockchain environment.

Hyperledger Fabric: Steps to Register, Authorize, and Manage Users

Hyperledger Fabric is a **permissioned blockchain** where access control is enforced by Certificate Authorities (CAs) and Membership Service Providers (MSPs).

Step-by-Step:

1. **Set Up Certificate Authority (CA):**
    - Deploy a Fabric CA to manage digital identities (X.509 certificates).
    - Each organization in the network typically runs its own CA.
2. **Register Users:**
    - Admin registers a new user with the CA.
    - CA creates an enrollment ID and secret for that user.
3. **Enroll Users:**
    - The user uses the ID and secret to enroll and get a digital certificate.

4. **Add User to MSP:**
    - The certificate is stored under the MSP directory, allowing the peer or client to authenticate as that user.
5. **Assign Roles:**
    - Roles can be assigned (e.g., client, admin, peer) using attributes embedded in certificates.
6. **Smart Contract (Chaincode) Logic:**
    - Access rights can be coded directly into smart contracts (chaincode) using attribute-based access control (ABAC).

Private Channels or Smart Contracts Limit Data Visibility

1. **Private Channels:**
    - A **channel** is a sub-network of the blockchain.
    - Only specific organizations (peers) have access to a channel.
    - Each channel maintains a **separate ledger and smart contract**.

    Example: Finance department and Auditors share a private channel; others can't access their data.

2. **Private Data Collections:**
    - Store **sensitive data privately** on specific peers, while only the hash is shared on the channel ledger.
    - Ensures **confidentiality without compromising integrity**.
3. **Smart Contracts (Chaincode) with Access Control:**
    - You can define **role-based access control** in the contract logic.
    - E.g., Only users with a role “manager” can approve transactions over a certain limit.