LAB01 - BLOCKCHAIN

Danh sách thành viên:
1. 22C11002 - Ngô Minh Đức
2. 22C11030 - Đặng Văn Hiển
3. 22C11035 - Đặng Trung Kiên

1. Cấu Trúc Dự Án:

- Cấu trúc dự án Blockchain sử dụng ngôn ngữ Go (Golang). Dự án này bao gồm cấu trúc dữ liệu cho giao dịch, khối (block), và chuỗi khối (blockchain), cũng như các chức năng để tạo mới khối, thêm giao dịch, in ra thông tin của blockchain, kiểm tra sự tồn tại của giao dịch trong blockchain
- Dự án được chia làm hai package là package main và package blockchain
  Trong đó:
  + package main có trách nhiệm tạo blockchain, thêm giao dịch và in ra màn hình.
  + package blockchain có trách nhiệm định nghĩa cấu trúc Transaction, Block, Blockchain và một số các hàm tạo Block, tinh Hash và một số các hàm tính toán và quản lý các Block và Blockchain. Quản lý sự tồn tại của các transaction bằng việc sử dụng Merkle Tree.
- Mục tiêu chính của bài lab này là xây dựng một blockchain đơn giản. Blockchain là một cơ sở dữ liệu phân tán của các ghi chú, và điều làm nó độc đáo chính là việc sử dụng hàm băm mật mã để tạo ra một cơ chế bảo mật chống thay đổi thông qua sự đồng thuận phân tán. Trong phần lớn trường hợp, blockchain là không cần phải xin phép, có nghĩa là nó cho phép sự tham gia công khai của các nút, thường được triển khai trên một mạng ngang hàng, tạo ra một cơ sở dữ liệu phân tán công cộng. Blockchain đã làm cho tiền điện tử và hợp đồng thông minh trở nên khả thi

2. Cấu Trúc Dữ Liệu

2.1 Transaction:

+ ID: String
+ Data: Dãy byte chứa dữ liệu của giao dịch.

2.2 Block: Trong blockchain các khối chứa thông tin quan trọng. Mỗi khối không chỉ lưu trữ các giao dịch mà còn chứa thông tin kỹ thuật như phiên bản hiện tại, timestamp, và hash của khối trước đó. Timestamp là thời điểm hiện tại (khi khối được tạo), Transactions chứa thông tin quan trọng trong khối, PrevBlockHash lưu trữ hash của khối trước đó, và Hash là hash của khối. Để tính toán hash SHA-256 của dữ liệu, hàm SetHash được sử dụng. Để tính toán checksum SHA-256 của dữ liệu, hàm Sum256 từ gói crypto của Golang được sử dụng.

+ Timestamp: Thời điểm tạo block theo định dạng Unix timestamp.
+ Transactions: Danh sách con trỏ đến các đối tượng Transaction.

+ PrevBlockHash: Dãy byte đại diện cho hash của block trước đó.
+ MerkleRoot: Dãy byte đại diện cho Merkle root của các giao dịch trong block.
+ Hash: Dãy byte đại diện cho hash của block hiện tại.

2.3 Blockchain: Một blockchain có thể được hiện thực bằng cách sử dụng một mảng và một bản đồ: mảng giữ các hash theo thứ tự chèn và bản đồ giữ các cặp hash và khối (do mảng không theo thứ tự). Để tiện lợi trong bản mẫu blockchain của bạn, bạn chỉ cần sử dụng một mảng như sau:

+ blocks: chứa con trỏ đến các đối tượng Block.

3. Nhận Xét về Thiết Kế và Thực Hiện

3.1 Blockchain và Cấu Trúc Danh Sách Liên Kết

Blockchain được triển khai thông qua cấu trúc danh sách liên kết, mỗi block có con trỏ đến block trước đó, tạo nên một chuỗi khối liên kết.

Điều này giúp đảm bảo tính toàn vẹn của dữ liệu trong chuỗi khối và dễ dàng xác minh lịch sử giao dịch từ genesis block đến block hiện tại.

3.2 Biểu Diễn Giao Dịch (Transaction)

Giao dịch được biểu diễn thông qua cấu trúc dữ liệu sử dụng dãy byte, điều này tạo ra sự linh hoạt trong việc đối mặt với nhiều loại dữ liệu khác nhau. Điều này quan trọng khi xây dựng hệ thống blockchain có thể xử lý nhiều loại thông tin giao dịch.

3.3 Cây Merkle và Xác Minh Giao Dịch

Cây Merkle được hiện thực một cách hiệu quả để tính toán Merkle root. Các giao dịch được biểu diễn dưới dạng các phần tử lá trong cây Merkle, giúp xác minh tính toàn vẹn của dữ liệu trong một block.

Việc kiểm tra tính toàn vẹn này thông qua Merkle root giúp trong quá trình xác minh giao dịch mà không cần phải xem xét toàn bộ nội dung của block.

3.4 Điểm Mạnh

Thiết kế sử dụng con trỏ trong cấu trúc danh sách liên kết giúp trong việc duyệt qua chuỗi khối một cách hiệu quả.

Việc sử dụng dãy byte để biểu diễn giao dịch tạo ra tính linh hoạt và có thể mở rộng cho các loại dữ liệu khác nhau.

3.5 Điểm Cần Cải Thiện

Cần thêm các chức năng bảo mật để đảm bảo tính an toàn và không thể thay đổi của dữ liệu trong blockchain.

Việc xử lý ngoại lệ và các trường hợp đặc biệt nên được xem xét để tăng tính ổn định của hệ thống.

3.6 Tổng Kết

Dự án đạt được tính chấp nhận và linh hoạt trong việc xử lý giao dịch, tính toàn vẹn của dữ liệu được đảm bảo thông qua sự tích hợp của cây Merkle. Để cải thiện, cần tập trung vào các khía cạnh bảo mật và xử lý các tình huống đặc biệt để tạo ra một hệ thống blockchain hoàn chỉnh và an toàn.

4. Hoạt Động của Hệ Thống

4.1 Khởi Tạo Blockchain

Quá trình khởi tạo blockchain bắt đầu với việc tạo một block genesis, thường chứa thông tin như timestamp, danh sách giao dịch và hash trước đó.
Hàm NewBlockchain được sử dụng để tạo blockchain mới với một block genesis.

4.2 Thêm Giao Dịch

Mỗi khi có giao dịch mới, một block mới được tạo ra sử dụng hàm AddBlock.
Giao dịch này được thêm vào danh sách giao dịch của block, và block mới này được liên kết với block trước đó thông qua con trỏ PrevBlockHash.

4.3 In Ra Blockchain

Hàm PrintBlockchain được sử dụng để in ra màn hình toàn bộ thông tin của các block trong blockchain.
Điều này giúp kiểm tra lịch sử giao dịch và theo dõi cấu trúc của blockchain.

4.4 Xác Minh Giao Dịch với Cây Merkle: Trong hàm VerifyTransactionInMerkleTree, tính toàn vẹn của một giao dịch được kiểm tra bằng cách so sánh Merkle root của block chứa giao dịch đó với Merkle root tính toán từ tất cả các giao dịch trong block. Điều này đảm bảo rằng giao dịch nằm trong cây Merkle của block đó.

4.5 Tổng Kết Hoạt Động

Blockchain không chỉ là nơi lưu trữ giao dịch mà còn là một hệ thống đảm bảo tính minh bạch và an toàn.
Hệ thống này sử dụng cấu trúc dữ liệu dạng danh sách liên kết để tạo ra một chuỗi các block một cách có tổ chức.

5. Thách Thức Trong Quá Trình Thực Hiện

- Lựa Chọn Cấu Trúc Dữ Liệu: Quyết định về biểu diễn của giao dịch và đảm bảo tính tương thích với cấu trúc cây Merkle.
- Quản Lý Con Trỏ: Quản lý đúng con trỏ để tránh rò rỉ bộ nhớ và đảm bảo liên kết chính xác giữa các block.
- Thách thức nổi bật là đảm bảo tính an toàn và bảo mật của blockchain trước các mối đe dọa như tấn công 51%, tấn công kéo dài và lỗ hổng bảo mật.
- Quá trình này đòi hỏi sự hiểu biết sâu rộng về lý thuyết mật mã và các chiến lược đảm bảo an ninh.

6. Bài Học Học Được

- Khái Niệm Mật Mã: Hiểu về cách sử dụng mã hóa hash (như SHA-256) trong blockchain giúp bảo vệ tính toàn vẹn của dữ liệu. Điều này không chỉ đặt ra vấn đề về bảo mật mà còn liên quan đến khả năng xác minh giao dịch và tính nhất quán của chuỗi khối.

- Xử Lý Con Trỏ: Nắm vững kỹ thuật xử lý con trỏ trong Go, đặc biệt là trong việc quản lý cấu trúc danh sách liên kết, là một bài học quan trọng. Quản lý chúng một cách đúng đắn không chỉ đảm bảo tính nhất quán của dữ liệu mà còn tránh được những vấn đề về hiệu suất và bộ nhớ.

- Thiết Kế Modul: Việc hiểu và thực hành thiết kế modul trong dự án blockchain giúp tạo ra một cấu trúc tổ chức, dễ bảo trì và mở rộng. Điều này đặc biệt quan trọng khi xây dựng các hệ thống lớn và phức tạp như blockchain, nơi sự tổ chức có thể giảm thiểu rủi ro lỗi và tăng khả năng mở rộng.

7. Tham Khảo

- https://go.dev/doc/
- https://viblo.asia/p/tuong-tac-voi-smartcontract-tren-blockchain-voi-go-phan-1-1Je5EjD4KnL
- https://academy.binance.com/vi/articles/double-spending-explained
- https://academy.binance.com/vi/articles/what-is-public-key-cryptography
- https://insider.blockchainwork.net/merkle-tree-cay-merkle-la-gi#:~:text=Blockchain%20l%C3%A0%20m%E1%BB%99t%20s%E1%BB%95%20c%C3%A1i,Tree%E2%80%9D%20(C%C3%A2y%20Merkle).
- https://academy.binance.com/vi/articles/what-is-public-key-cryptography
-