

区块链演示项目技术方案

一、项目概述

本项目是一个简单的区块链演示项目，结合了 Proof of History (POH) 和 Delegated Proof of Stake (DPoS) 共识机制，用于模拟区块链的基本操作，如区块创建、添加交易和计算哈希值。

二、技术架构

1. 核心数据结构

- Block 结构体**：表示区块链中的一个区块，包含区块索引、时间戳、哈希值、领导节点、奖励、交易数据和前一个区块的哈希值。
- Blockchain 结构体**：表示整个区块链，由一个区块向量组成。
- POH 结构体**：实现 Proof of History 共识机制，通过计数器记录事件。
- DPoS 结构体**：实现 Delegated Proof of Stake 共识机制，管理验证者列表和当前验证者索引。

2. 核心功能

- 区块哈希计算**：使用 `Sha3_256` 哈希算法计算每个区块的哈希值。
- 区块创建**：根据给定的索引、交易数据和前一个区块的哈希值创建新的区块。
- 区块链初始化**：创建包含一个创世区块的区块链。
- 添加区块**：将新的区块添加到区块链中。
- 事件记录**：使用 POH 结构体记录事件。
- 验证者选择**：使用 DPoS 结构体选择当前验证者。

三、代码实现

1. 依赖库

- `chrono`：用于处理时间戳。
- `sha3`：用于计算哈希值。

2. 核心代码逻辑

- Block 结构体实现**：
 - `calculate_hash` 方法：计算区块的哈希值。
 - `new` 方法：创建新的区块。
- Blockchain 结构体实现**：
 - `new` 方法：创建新的区块链。
 - `get_latest_block` 方法：获取区块链中的最后一个区块。
 - `add_block` 方法：将新的区块添加到区块链中。
- POH 结构体实现**：
 - `new` 方法：创建新的 POH 实例。
 - `record_event` 方法：记录一个事件。
- DPoS 结构体实现**：
 - `new` 方法：创建新的 DPoS 实例。

- `get_current_validator` 方法：获取当前验证者。
- **main 函数：**
 - 创建区块链、POH 实例和 DPoS 实例。
 - 添加一些区块到区块链中。
 - 打印区块链中的所有区块信息。

四、项目运行步骤

1. 确保已经安装了 Rust 开发环境。
2. 克隆项目代码到本地：

```
git clone <项目仓库地址>  
cd blockChainDemo2
```