

项目说明

1. 项目目标

1.1 核心目标

- 完成一个简单的区块链系统，能够生成创世区块并通过程序逻辑添加新区块。
- 输出区块链信息，包括区块哈希值、前一个区块的哈希值、时间戳等。
- 区块添加逻辑：通过程序内部逻辑自动添加区块，区块数据为简单的字符串。

1.2 简化功能

- 聚焦核心功能：确保区块链的基本数据结构（区块、区块链）和核心逻辑（创世区块生成、区块添加）能够正常运行。

2. 开发计划

2.1 时间安排

- 第 1 天：完善区块和区块链的现有代码，确保创世区块生成和区块添加功能正常。
- 第 2 天：实现程序内部的区块添加逻辑，支持自动添加多个区块。
- 第 3 天：编写单元测试，确保区块和区块链的核心逻辑正确。
- 第 4 天：优化代码结构，增加注释，提高代码可读性。
- 第 5 天：撰写项目文档，准备演示。

2.2 风险与应对

1. 时间不足:

- **应对措施:** 优先实现核心功能（创世区块生成、区块添加、区块链展示），其他功能（如 Merkle 哈希计算）可以简化或延后。

2. 代码质量问题:

- **应对措施:** 编写单元测试，确保核心逻辑正确；代码结构尽量简洁清晰。

3. 项目文档

3.1 使用说明

1. 克隆项目:

```
git clone https://github.com/littleeye/blockchain_rust.git
cd blockchain_rust
```

2. 运行项目:

```
cargo run
```

4. 后续扩展

如果时间允许，可以在现有基础上逐步添加以下功能：

1. **工作量证明（PoW）**：实现简单的挖矿功能。
2. **交易模拟**：支持模拟交易并打包到区块中。
3. **网络通信**：实现简单的 P2P 网络通信，支持多节点同步区块链。

5.运行截图

```
produce a block!

start mining ...
produce a block!
+++++
Block {
  header: BlockHeader {
    time: 1741424515,
    tx_hash: "919c913797a6832bc8176dddb9b33abb2e244bcf1eea2bb81b8e5ad939ed2ce9",
    pre_hash: "",
  },
  hash: "3e417d7335898b8c14df94c667c61d2e81597f9cc8188c5287f6730ebc34033f",
  data: "This is genesis block",
}

+++++
Block {
  header: BlockHeader {
    time: 1741424520,
    tx_hash: "f3ffbf3adad0f5320f5205df4f65d6c9f56f1fd55d9194ab44ab6f6da3961de8",
    pre_hash: "3e417d7335898b8c14df94c667c61d2e81597f9cc8188c5287f6730ebc34033f",
  },
  hash: "97d7ef96f53d2db7f7b42348ab50b5ecca4d58ea7b8e4da6c82d970dfbae02b9",
  data: "a -> b: 5 newcoin",
}

+++++
Block {
  header: BlockHeader {
    time: 1741424525,
    tx_hash: "fbca4ce9f15cd395e50f78598cdea37db0f4c8cb30d786db0d55078fd0863a48",
    pre_hash: "97d7ef96f53d2db7f7b42348ab50b5ecca4d58ea7b8e4da6c82d970dfbae02b9",
  },
```

问题 输出 终端 端口 调试控制台

```
hash: "1836d68937f197f661b6f9fb9c67b9f5c56ed31bd19efa9d47805ef8d6a6f109",
data: "This is bolck 9",
}

start mining ...
produce a block!
+++++
Block {
  header: BlockHeader {
    time: 1741424575,
    tx_hash: "6032e267b35ffd30fb2b18a4f3d7f19f4a728c37951e00316f86d4d9ad18b689",
    pre_hash: "1836d68937f197f661b6f9fb9c67b9f5c56ed31bd19efa9d47805ef8d6a6f109",
  },
  hash: "31b194be4e9b6d8d3ad2a21bc91dc922488444ee3a2890653f447a192a33654d",
  data: "This is bolck 10",
}

start mining ...
produce a block!
+++++
Block {
  header: BlockHeader {
    time: 1741424580,
    tx_hash: "2ae50659a8e3f0324e853ff9015fbcf582c336b06ff5c91eddfd703c5ffb8282",
    pre_hash: "31b194be4e9b6d8d3ad2a21bc91dc922488444ee3a2890653f447a192a33654d",
  },
  hash: "52cd33279c2744979991f87885b923a3cbedd23fec550823bd72586ac7376bb4",
  data: "This is bolck 11",
}

start mining ...
produce a block!
+++++
```