

实验1 - 区块链基本结构

1. 实验内容

1.1 实现一个简易的Merkle Tree (20')

1.2 生成若干个比特币账户 (5')

1.3 利用 1.2 中生成的账户，对消息进行签名和验证 (5')

1.4 利用 1.2 中生成的账户，生成自定义交易 (25')

1.5 利用 1.1 中MerkleTree功能，将1.4中生成的交易打包成多个区块以链式组织起来 (25')

2. 实验要求

2.1 Merkle Tree

- 生成一个MerkleTree类，该类需要实现基本的make_merkle_tree方法;
- make_merkle_tree方法
 - 输入: 一个字符串的迭代器 (iterator of arbitrary strings)
 - 输出: 一颗MerkleTree
 - 计算Hash的过程Bitcoin一致，涉及到的Hash方法可借助相关包
 - 该方法需要应保证确定性，即对于相同的输入，多次运行的结果应保持一致
 - 不限定数据结构，但尽可能高效
- 除make_merkle_tree方法外，可自定义实现其他方法

2.2 比特币账户

- 生成100个比特币账户，每个比特币账户包含3部分，分别为私钥，公钥和地址
- 具体实现过程可参见《Mastering Bitcoin》相关章节

2.3 签名及其验证

- 根据椭圆曲线数字签名算法ECDSA，利用1.2生成的私钥和公钥，对字符串消息"blockchain-ss-2021"进行签名
- 验证上一步的签名是否正确

2.4 交易的生成

- 实现一个方法，用于随机生成1笔交易，该方法的输入输出请自行设计
- 交易脚本类型为P2PKH，签名哈希类型为SIGHASH ALL
- 交易的版本（Version）设置为1，锁定时间（Locktime）设置为全0
- 交易输入的Sequence设置为全F
- 利用上述方法和1.2生成的100个账户，随机生成1000个交易
- 交易的具体结构可参见《Mastering Bitcoin》相关章节

2.5 区块的生成

- 生成10个区块，每个区块包含100笔交易，交易来源为1.4中生成的1000笔交易，不需要考虑Coinbase交易和Segregated Witness
- 区块的具体结构可参见《Mastering Bitcoin》相关章节
- 生成的第1个区块中，Prev Hash设置为全0
- 生成的区块中，区块头中的Timestamp，Target，Nounce均设置为全0
- 为了验证所实现算法的正确性，请先利用真实的交易数据和区块数据进行测试，可借助比特币浏览器获取相关数据，建议使用高度在10000之前的区块

3. 提交内容


3.1 代码文件

- 请编写可读性良好的代码，并撰写相关注释
- 将所有代码文件打包为zip压缩文件

3.2 实验报告

- 说明相关的数据结构和算法设计
- 自定义测试数据，验证自定义的算法需验证其正确性

4. 提交时间

- 请于2021年10月31日前将提交内容发送至课程邮箱 

5. 其他

- 不限制实现语言，可借助开源库实现
- 注意交易和区块中的字节序问题
- 评分标准分为2部分
 - 实验完成度
 - 代码和实验报告质量
- 参考资料
 - <https://github.com/tianmingyun/MasterBitcoin2CN>
 - <https://bitcoin.stackexchange.com/questions/72657/signature-verification-in-python-using-compressed-public-key>
 - <https://medium.com/@bitaps.com/exploring-bitcoin-signing-the-p2pkh-input-b8b4d5c4809c>
 - <https://www.jianshu.com/p/a560e0605ff2>