

1. 在非对称加密算法和数字签名的过程中，私钥和公钥的作用是否相同？若不同，请描述不同之处。

发起人私钥签名，发起人公钥验签

发起方对包用自己的私钥签名，接收方收到包后用发送方的公钥验签

私钥：识别发送方是否是合法的

非对称加密

接收人公钥加密，接收人私钥解密

发起方使用接收方的公钥进行加密，接收方用自己的私钥进行解密

公钥：保证加密后接收方能够解密

1. UTXO模型和账户模型的各自优缺点是什么？

UTXO(Unspent Transaction Output)

1. 因为丢失私钥丢失账户控制权后，UTXO会一直保存账户的余额,因而没有输出.
2. 验证一笔交易的余额是否足够需要向上追溯，除了BTC交易外每一笔交易的输出都是另一笔交易的输入
3. 大量生成地址，每个地址只使用一次，一旦该地址付出过比特币，那么公钥就暴露了，也就不抗量子攻击了，所以找零不会回到付款地址，而是一个新地址

优点：

1. 只有直到私钥才有账户的控制权
2. 可以一定程度上避免双花攻击
3. 通过找零时设置新地址增加了一定稳定性
4. 在简单业务和跨链上，UTXO 有其非常独到和开创性的优点

缺点：

1. 丢失账户控制权后，UTXO会一直保存账户的余额；随着比特币的分支和碎片化以及账户私钥丢失，UTXO模型会越来越膨胀
2. 交易时效率可能受到限制，交易成本较高
3. 没有一个相对稳定的身份，一旦出现交易失败或合约违约，无法断定是谁的责任

Account模型

优点：

具有一个相对稳定的身份，可以支持智能合约

合约以二进制形式保存在Account中，并且Account拥有自身状态。模型具有更好的可编程性

Account 模型在可编程性，灵活性等方面更有优势

批量交易成本低

缺点：

模型交易之间没有依赖性，需要解决重放问题

对于实现闪电网络等，用户需要更复杂的证明机制

是一种状态机。交易是事件本身，不包含结果。

3.

在实验完成的基础上，在test/java/unit包中新建一个测试类UtxoTest，构造一个确定的UTXO交易：账户A(accounts[1])转账1000BTC 给账户B(accounts[2])。

