1、如何用非对称密码体制实现双向认证？请叙述主要过程。

2、数字签名应具有哪些性质？

3、直接数字签名和仲裁数字签名的区别？

4、

在数字签名标准（DSS）中，假设p=83，q=41，h=3。求

（1）参数g；

（2）取私钥x=57，求公钥y；

（3）设消息M=56，取随机数k=23，求M的签名；（注：为了简化，用M代替SHA（M））

（4）对M=56的上述签名进行验证。

实现双向认证主要包括以下步骤：

（1）客户端A发送自己的公钥给服务器B。

（2）服务器B发送自己的公钥给客户端A。

（3）客户端A使用服务器B的公钥加密一个随机数并发送给服务器B。

（4）服务器B使用自己的私钥解密收到的消息，并使用客户端A的公钥加密同样的随机数并发送给客户端A。

（5）客户端A使用自己的私钥解密服务器B发来的消息，如果得到的随机数与之前发出的相同，则表明服务器B是可信的。

数字签名应具有以下性质：

（1）验证方可以确定签名是否由合法的签名者产生。

（2）签名者不能否认曾经创建过签名。

（3）签名者不能否认文件内容在签名时没有被修改过。

（4）签名结果无法复制或转移。

（5）存储签名结果需要耗费足够的代价。

（6）签名算法具有可证明的安全性。

直接数字签名和仲裁数字签名的区别主要在于签名的方式不同。直接数字签名是指签名者直接对原始数据进行签名，而仲裁数字签名是指签名者对原始数据及其摘要进行签名。

根据DSS标准：

（1）参数g = h ^ ((p - 1) / q) mod p = 3 ^ ((83 - 1) / 41) mod 83 = 74。

(2) 公钥y = g ^ x mod p = 74 ^ 57 mod 83 = 40。

(3) 计算r：r = (g ^ k mod p) mod q = (74 ^ 23 mod 83) mod 41 = 29；

计算s：s = (k^-1 \* (M + xr) mod q = (23^-1 \* (56 + 5729) mod 41 = 36。

因此，M的数字签名为（r,s）= (29, 36)。

(4) 计算w：w = s ^ -1 mod q = 36^-1 mod 41 = 31；

计算u1：u1 = (w\*M mod q) = (31 \* 56) mod 41 = 9；

计算u2：u2 = (w\*r mod q) = (31 \* 29) mod 41 = 28；

计算v：v = ((g ^ u1 \* y ^ u2 mod p) mod q = ((74 ^ 9 \* 40 ^ 28) mod 83) mod 41 = 29；