1.哈希函数是一个数学函数，有以下三个特性：

其输入可为任意大小的字符串

产生固定大小的输出

能进行有效运算，对于特定的输入字符串，在合理时间内，可以计算出哈希函数的输出。对应n位的字符串，其哈希值计算的复杂度为O(n).

加密的哈希函数附加特性：

碰撞阻力

这里的碰撞指不同的输入，相同的输出

如果无法找到两个值，x和y , x不等于y,而 H(x)=H(y), 则称哈希函数H具有碰撞阻力。

找不到碰撞，但是不表示不存在碰撞，且有些方法能保证找到碰撞

生日悖论（birthday paradox）:如果一个房间里有23个或23个以上的人，那么至少有两个人的生日相同的概率要大于50%。

对于某些哈希函数，无法确认识别碰撞的有效方法是否存在，只是怀疑这些函数具有防碰撞的特性，但是经过证明，世界上没有哈希函数具有防碰撞的特性。

碰撞阻力的应用：将哈希输出作为信息摘要（meassage digest）

隐秘性

哈希函数H具有隐秘性，如果：当其输入r选自一个高阶最小熵（high min-entroy）的概率分布，在给定H(r||x)条件下来确定x是不可行的。

在信息论中，最小熵是用于测试结果可预测性的手段，而高阶最小熵直观的描述了分布（如随机变量）的分散程度。

保证知道哈希函数的输出y=H(x),是没有可行的办法输出x的。

应用：承诺

谜题友好

如果对于n位输出值y,假定k选自高阶最小熵分布，如果无法找到一个可行的方法，在比2^n小很多的时间内找到x,保证H(k||x)=y成立，那么我们称哈希函数H为谜题友好。

应用：探索谜题

**安全哈希算法（Secure Hash Algorithm 256,SHA\_256）**

哈希指针（hash pointer）是一个指向数据存储位置及其位置数据的哈希值的指针。

通过哈希指针而不是普通指针构建的一个链表，我们把这个链表称为区块链。

使用哈希指针的二叉树也称为梅克尔树（Merkle trees）

所有的数据区块都被两两分组，指向这些数据区块的指针被存储在上一层的父节点（parent node）中，而这些父节点再次被两两分组，并且指向父节点的指针被存储在上一层的父节点中，一直持续这个过程，知道最后到达树的根节点。

保证了区块链的不 可篡改性

隶属证明

为了证明某个数据区块来自一个梅克尔树，我们只需要找到该数据区块到树根节点的路径。

如果整棵树上有n个节点，只需要展示log(n)个项目，因此每个步骤仅需要计算子区块的哈希值，验证过程需要时间log(n)。

非隶属证明

数字签名(digital signatures)

椭圆曲线数字签名算法（ECDSA）

个人密钥：256位

公钥（未压缩）：512位

公钥（压缩）：257位

待签名信息：256位

签名：512位