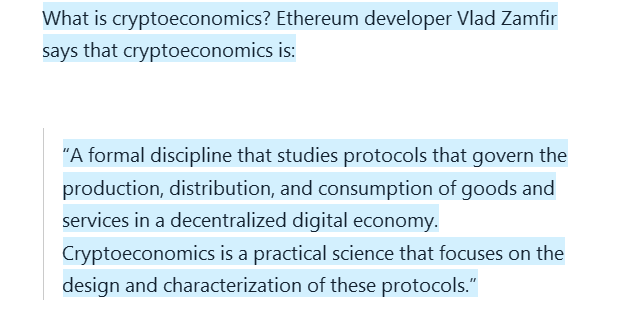
# Cryptoeconomics（加密经济）

## 是什么：

以太坊创始人的定义：

A formal 学科研究在去中心化数字经济中的商品的生产，分发，消耗的协议。

加密经济就是就是关于这些协议的设计与描述的实践的科学。



加密经济= 加密 + 经济。

## 基本特性（properties）

## Cryptography

动机（目的）

为什么需要应用加密技术？

## Economics

## 工作模型之区块链技术

区块链技术就是运行在加密经济的properties之上。

正是加密经济其中的“经济”使得区块链具有了独特的能力。

### 传统P2P网络：torrent技术的问题：

区块链并不是第一个去中心化的P2P系统。用于文件共享的Torrent技术就已经是并且应用了很多年多了。

Torrent技术的缺点：

Torrent系统内，任何人都可以通过去中心化的P2P网络共享文件，该技术的基本实现思路是，任何人通过从网络内下载文件时，同时也为网络内的其他人做种子，以便其他人也可以从本机上下载文件。

缺点：

Torront系统只有在具有分享精神有荣誉感的人之间才能更好的工作，因为实际情况是，很多人只是自私地从网络中下载文件，而拒绝为其他人做种子。

The problem was that this worked on an honor system. If you were downloading a file, then you were expected to seed as well. The problem is that humans are not really the most honorable of creatures and without any economic incentives it made no sense for people to keep seeding a file which took up unnecessary space in their computers.

# 区块链技术的创新：

1、激励机制的引入：在传统的去中心化网络中，加入了economic incentive，以激励人们“follow the rules”

2、共识机制的引入：共识机制是为了解决在去中心化的,相互独立的节点如何就某些事项达成一致的过程。简而言之，共识机制是在解决分布式系统中如何保持一致性的问题。

# Cryptoeconomic properties of BitCoin

* It is based on the blockchain technology where each block contains the hash of the previous block and forms a continuous chain.
* Each block will include transactions.
* The blocks will have a particular state which is subject to change according to transactions. Eg. if A has 50 bitcoins and wants to send 20 bitcoins to B. Then The new state should show that A has 30 bitcoins left and B has 20 new bitcoins.
* The blockchain must be immutable. It should be possible to add new blocks but the old blocks can’t be tampered with.
* Only valid transactions should be allowed.
* The blockchain should be downloadable and anyone anywhere can easily access and check a particular transaction.
* Transactions could be added quickly to the blockchain if a sufficiently high transaction fee is paid.

Cryptography：

# 概念术语

## 金融/会计基本概念

### financial statements:

（reference from wikipedia）

Financial statements (or financial report) is a formal record of the financial activities and position of a business, person, or other entity.

Relevant financial information is presented in a structured manner and in a form easy to understand. They typically include basic financial statements, accompanied by a management discussion and analysis:[2]

**A balance sheet or statement of financial position,** reports on a company's assets, liabilities, and owners equity at a given point in time.

**An income statement or statement of comprehensive income,** statement of revenue & expense, P&L or profit and loss report, reports on a company's income, expenses, and profits over a period of time. A profit and loss statement provides information on the operation of the enterprise. These include sales and the various expenses incurred during the stated period.

**A Statement of changes in equity or equity statement or statement of retained earnings**, reports on the changes in equity of the company during the stated period.

**A cash flow statement** reports on a company's cash flow activities, particularly its operating, investing and financing activities.

For large corporations, these statements may be complex and may include an extensive set of footnotes to the financial statements and management discussion and analysis. The notes typically describe each item on the balance sheet, income statement and cash flow statement in further detail. Notes to financial statements are considered an integral part of the financial statements.

### liability（负债）

wikipedia：

In financial accounting, a liability is defined as the future sacrifices of economic benefits that the entity is obliged to make to other entities as a result of past transactions or other past events,[1] the settlement of which may result in the transfer or use of assets, provision of services or other yielding of economic benefits in the future.

### assets（资产）

wikipedia：

In financial accounting, an asset is an economic resource. Anything tangible or intangible that can be owned or controlled to produce value and that is held by a company to produce positive economic value is an asset. Simply stated, assets represent value of ownership（所有权的价值体现） that can be converted into cash (although cash itself is also considered an asset).[1]

## 记账相关

### ledger（账本）

账本的作用：

传统账本技术：

每个市场参与者都维护（逻辑上的）一个账本，用来追踪自身的（某项）资产的增减变动（收入、支出）；用于展示核查资产状况；

wikipedia：

账本分三种类型：

debtors ledger：债务人账本（销售账本）记录了产生收益的交易，提供收入信息；

creitors ledger：债权人账本（支出账本），记录导致利益减少的交易，提供支出信息；

general leger：汇总账本，representing the five main[3][citation needed] account types: assets, liabilities, income, expenses, andCapital.

The ledger is a permanent summary of all amounts entered in supporting [journals](https://en.wikipedia.org/wiki/General_journal) which list individual transactions by date. Every transaction flows from a journal to one or more ledgers. A company's [financial statements](https://en.wikipedia.org/wiki/Financial_statement) are generated from summary totals in the ledgers.[[2]](https://en.wikipedia.org/wiki/Ledger#cite_note-2)

Ledgers include:

* [**Sales ledger**](https://en.wikipedia.org/wiki/Sales_ledger)**,** records [accounts receivable](https://en.wikipedia.org/wiki/Accounts_receivable). This ledger consists of the financial transactions made by customers to the company.
* [**Purchase ledger**](https://en.wikipedia.org/wiki/Purchase_ledger) records money spent for purchasing by the company.
* [**General ledger**](https://en.wikipedia.org/wiki/General_ledger) representing the five main[[3]](https://en.wikipedia.org/wiki/Ledger#cite_note-3)[[*citation needed*](https://en.wikipedia.org/wiki/Wikipedia:Citation_needed)] account types: [assets](https://en.wikipedia.org/wiki/Assets), [liabilities](https://en.wikipedia.org/wiki/Liability_(financial_accounting)), [income](https://en.wikipedia.org/wiki/Income), [expenses](https://en.wikipedia.org/wiki/Expenses), and[Capital](https://en.wikipedia.org/wiki/Financial_capital).

For every [debit](https://en.wikipedia.org/wiki/Debits_and_credits) recorded in a ledger, there must be a corresponding [credit](https://en.wikipedia.org/wiki/Debits_and_credits) so that the debits equal the credits in the grand totals.

The three types of ledgers are the general, debtors, and creditors.[[4]](https://en.wikipedia.org/wiki/Ledger#cite_note-4) The [**general ledger**](https://en.wikipedia.org/wiki/General_ledger)accumulates information from journals. Each month all journals are totaled and posted to the General Ledger. The purpose of the General Ledger is therefore to organize and summarize the individual transactions listed in all the journals. **The Debtor Ledger** accumulates information from the sales journal. The purpose of the Debtors Ledger is to provide knowledge about which customers owe money to the business, and how much. **The Creditors Ledger** accumulates information from the purchases journal. The purpose of the Creditors Ledger is to provide knowledge about which suppliers the business owes money to, and how much.

## 安全相关：

### Cryptographic hash function

参考：

<https://en.wikipedia.org/wiki/Cryptographic_hash_function>

A **cryptographic hash function** is a special class of [hash function](https://en.wikipedia.org/wiki/Hash_function) that has certain properties which make it suitable for use in [cryptography](https://en.wikipedia.org/wiki/Cryptography). It is a mathematical [algorithm](https://en.wikipedia.org/wiki/Algorithm) that [maps](https://en.wikipedia.org/wiki/Map_%28mathematics%29) data of arbitrary size to a [bit string](https://en.wikipedia.org/wiki/Bit_string) of a fixed size (a hash) and is designed to be a [one-way function](https://en.wikipedia.org/wiki/One-way_function), that is, a function which is [infeasible](https://en.wikipedia.org/wiki/Computational_complexity_theory#Intractability) to invert. The only way to recreate the input data from an ideal cryptographic hash function's output is to attempt a [brute-force search](https://en.wikipedia.org/wiki/Brute-force_search) of possible inputs to see if they produce a match, or use a [rainbow table](https://en.wikipedia.org/wiki/Rainbow_table) of matched hashes. [Bruce Schneier](https://en.wikipedia.org/wiki/Bruce_Schneier) has called one-way hash functions "the workhorses of modern cryptography".[[1]](https://en.wikipedia.org/wiki/Cryptographic_hash_function#cite_note-1) The input data is often called the *message*, and the output (the *hash value* or *hash*) is often called the *message digest* or simply the *digest*.

The ideal cryptographic hash function has five main properties:

* it is [deterministic](https://en.wikipedia.org/wiki/Deterministic_algorithm) so the same message always results in the same hash
* it is quick to compute the hash value for any given message
* it is [infeasible](https://en.wikipedia.org/wiki/Computational_complexity_theory#Intractability) to generate a message from its hash value except by trying all possible messages
* a small change to a message should change the hash value so extensively that the new hash value appears uncorrelated with the old hash value
* it is [infeasible](https://en.wikipedia.org/wiki/Computational_complexity_theory#Intractability) to find two different messages with the same hash value

#### 破解办法：

##### brute-force attack

In [computer science](https://en.wikipedia.org/wiki/Computer_science), **brute-force search** or **exhaustive search**, also known as **generate and test**, is a very general [problem-solving](https://en.wikipedia.org/wiki/Problem-solving) technique and [algorithmic paradigm](https://en.wikipedia.org/wiki/Algorithmic_paradigm) that consists of systematically enumerating all possible candidates for the solution and checking whether each candidate satisfies the problem's statement.

While a brute-force search is simple to [implement](https://en.wikipedia.org/wiki/Software_implementation), and will always find a solution if it exists, its cost is proportional to the number of candidate solutions – which in many practical problems tends to grow very quickly as the size of the problem increases ([combinatorial explosion](https://en.wikipedia.org/wiki/Brute-force_search#Combinatorial_explosion))[[1]](https://en.wikipedia.org/wiki/Brute-force_search#cite_note-1). Therefore, brute-force search is typically used when the problem size is limited, or when there are problem-specific [heuristics](https://en.wikipedia.org/wiki/Heuristic_%28computer_science%29) that can be used to reduce the set of candidate solutions to a manageable size. The method is also used when the simplicity of implementation is more important than speed。

##### Rainbow table

参考：<https://en.wikipedia.org/wiki/Rainbow_table>

A **rainbow table** is a [precomputed](https://en.wikipedia.org/wiki/Precomputed) [table](https://en.wikipedia.org/wiki/Lookup_table) for reversing [cryptographic hash functions](https://en.wikipedia.org/wiki/Cryptographic_hash_function), usually for cracking password hashes. Tables are usually used in recovering a [password](https://en.wikipedia.org/wiki/Password) (or credit card numbers, etc.) up to a certain length consisting of a limited set of characters. It is a practical example of a [space–time tradeoff](https://en.wikipedia.org/wiki/Space%E2%80%93time_tradeoff), using less computer processing time and more storage than a [brute-force attack](https://en.wikipedia.org/wiki/Brute-force_attack) which calculates a hash on every attempt, but more processing time and less storage than a simple [lookup table](https://en.wikipedia.org/wiki/Lookup_table) with one entry per hash. Use of a [key derivation function](https://en.wikipedia.org/wiki/Key_derivation_function) that employs a [salt](https://en.wikipedia.org/wiki/Salt_%28cryptography%29) makes this attack infeasible.

Rainbow tables were invented by Philippe Oechslin[[1]](https://en.wikipedia.org/wiki/Rainbow_table" \l "cite_note-ophpaper-1) as an application of an earlier, simpler algorithm by [Martin Hellman](https://en.wikipedia.org/wiki/Martin_Hellman).[[2]](https://en.wikipedia.org/wiki/Rainbow_table#cite_note-:0-2)

##### a simple lookup table

## 区块链：

### Blockchain:

The blockchain is a chain of blocks where each block contains data of value **without any central supervision**. It is cryptographically secure and immutable.

区块链，每个区块中包含一些数据 of vlaue（具体应用场景不同，数据不一样）。整个区块链内的数据是密码学安全（只是密码学安全，不代表物理或其他安全）和不可修改的，另外整个区块链的运行维护的不依赖于一个中央控制节点。

### Decentralized

去中心化的。系统的运行维护不依赖于中央控制节点。

### Concensus Mechanism

共识机制。在去中心化的系统中，各个节点之间就某些事项达成一致的机制。

在中心化系统中，各个节点之间的交互往往依赖中心节点提供的某些服务基础上的。

### Miners

在区块链技术中，所有参与新的合法区块产生的节点（这些节点一般需要消耗某些资源才能产生新的区块）

### 创世区块（genesis block）

The genesis block is the first block of the blockchain, and the reason why it is special is that while every bock points to the block previous to it, the genesis block doesn’t point at anything.  So, the moment a new chain is created, the genesis block is invoked immediately.

# 背景动机目的（需求）

区块链技术

# 区块链技术的难点：（需求：系统需满足的要求）

## 1、保证安全性（Security）

区块链代码首先是开源的，所有人都可以看到，与一般开源项目不同，区块链项目代码中发现的漏桶很有可能会被黑客利用，而导致巨大的经济损失。正因为如此，所以区块链项目的开发一般进展比较慢。

## 2、资源管理（resource management）

it is important to keep pace with the network. You cannot fall too far behind and not keep up with all the network demands. You should be well equipped to handle remote and local queries

## 3、性能（performance）

区块链需要总是工作在最高的性能条件下，为此所选择的开发语言must be extremely versatile（万能的）。在区块链中，有些任务可以并行运行，而有些任务则必须顺序运行。

例如，其中一种可以并行运行的任务就是，数字签名的校验（校验的过程中所需要的输入只有三种：KEY,transaction,and signature）

而必须顺序执行的任务，则例如transaction execution （交易不能同时并行运行，一个时刻只能有一个交易运行，以避免类似double spending这种问题）。

有些语言擅长于并行运算，而有些语言擅长于非并行运算。

## 4、隔离（isolation）

问题需求：

在区块链开发中，所有的交易（transaction）都必须是确定的（deterministic）。也就是说，一个交易（transaction）不能behaves one way and then behaves another way the next day. 类似的在智能合约中，you cannot have smart contracts that work in two different ways on two different machines.

deterministic behavior的定义:

If A + B = C, then no matter what the circumstances, A+B will always be equal to C. That is called deterministic behavior.区块链技术中所使用的hash运算是确定性的功能（Hash functions are deterministic, meaning A’s hash will always be H(A).）。

解决方法：isolation

The only solution to this is isolation. Basically, you isolate your smart contracts and transactions from non-deterministic elements.（将区块链内要求严格确定性的功能与非确定性的功能分隔开）

开发语言选择：

There are some languages which fulfill most of these needs. If you are a blockchain developer, then you definitely need to have some basic knowledge of C++ and JavaScript.

While C++ may seem a little outdated, the truth is that it wonderfully satisfies all the functionalities that we have described above. In fact, Satoshi Nakamoto wrote the Bitcoin source code in C++.

Along with HTML and CSS it is one of the three core technologies in World Wide Web Content Production. Javascript is usually used to create highly interactive web pages.

# 区块链技术的组成：（满足了什么需求）

## 密码学技术：Cryptography

### Hashing：

hashing means taking an input string of any length and giving out an output of a fixed length. Bitcoin uses SHA-256 to take in an input string of any length and giving an out hash of 256 bits。

给定输入，总是产生特定的输出，而且从输出无法反推出输入；

问题：有完美哈希吗？

#### 在加密币中的应用：

1 Cryptographic hash functions.

2 Data structures.

3 Mining

##### Cryptographic hash functions

在区块链中应用的是一类称为Cryptographic hash的特殊hash函数。（详情参考上文概念术语部分）。

基本特性：

* **Deterministic（输出确定）:** An input A will always have the same output h(A) no matter how many times you parse it through the same hash function.
* **Quick Computation**: A function should return a hash of an input as quickly as possible.
* **Pre-Image resistance（单向）:** Given h(A) which is an output of a hash function, it should be infeasible to determine input A.
* **Collision resistance（无冲突）**: Given two inputs A and B and their hash outputs h(A) and h(B) it should be infeasible for h(A) = h(B).
* **Small changes（蝴蝶效应）**: in the input should drastically affect the output of the hash function.
* **Puzzle Friendly:** For every hash output Y and an input x. It is infeasible to find a value k, which will result in h(k|x) = Y

##### Data structures’不可篡改性，immutability

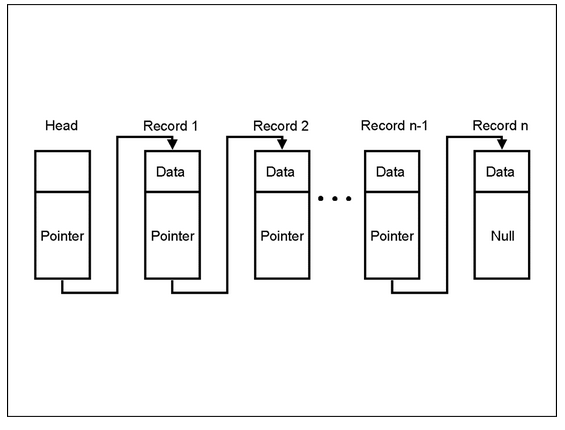
Hash的“Small changes”特性的为区块链的“immutability”特性提供了重要保障。

在区块链中，这一特性的应用就是：在区块链结构中的hash pointer的设计。

在区块链中有两个结构对于理解区块链比较重要：

Linked Lists and Hash Pointers。

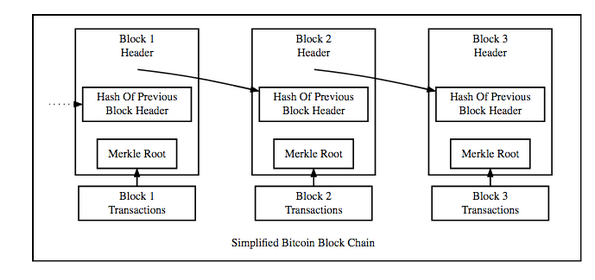
* **Linked Lists:** Linked lists are blocks of data which are connected to one after another. This is an example of a linked list:



Each block in the list is pointing to the other via a pointer.

* **Pointer:** Pointers are variables which include the addresses of the other variables. So they are variables which are literally pointing towards the other variables.
* **Hash Pointers:** Hash pointers are basically pointers which not only has the address of other variables but also the hash of the data in that variable. So how does that help in the context of a blockchain? （这是hash指针提供了区块链的不可篡改的特性，因为一旦某个区块被串改，会导致后一个区块保存的该区块的hash值与串改后的区块的hash值不一致，从而发现该区块被窜改过，被认定为不合法区块，所在的链也不合法。所以对于对于攻击者而言如果意图修改某一个区块链的内容，就需要保证将该区块之后的所有区块都修改（主要指每个区块中的hash值），而区块链的共识机制，hash等的设计导致这需要很大的资源消耗，所以攻击几乎不可能。

**This is what a blockchain looks like:**



The blockchain is basically a linked list where each new block contains a hash pointer which points to the previous block and the hash of all the data in it. Just this one property leads into one of Blockchain’s greatest qualities….its immutability

##### Mining

挖矿的过程就是各节点（或矿工节点）通过一种消耗资源（如时间，计算资源，电力等）方式计算出新的合法区块的过程。（至少在比特币中）这个过程是各个节点（或矿工节点）通过计算一个密码谜题来实现的，只有产生算法认可的计算结果时，该节点产生新的区块才被认为是合法的。**Hash在这个过程中的作用是**：挖矿过程中先设置一个难度系数（difficulty level），然后随机产生一个字符串称为“噪声”，“噪声”is appended to the hash of the new block and hashed again，然后检查最新计算出的hash值的特点(例如比特币中要求hash值的前n位必须是0)是否满足小于设置的难度系数，如果小于，则新区块被认为是合法的，可以加入到区块链中，然后系统会给矿工一定的奖励；如果不小于，则矿工继续产生新的“噪声”继续上面的计算过程。

### 数字签名：

数字签名是密码学技术的一种常见的应用，在现有的互联网、金融等已经广泛引用。其实现是基于密码学中的非对称加密技术，

数字签名三种基本功能：

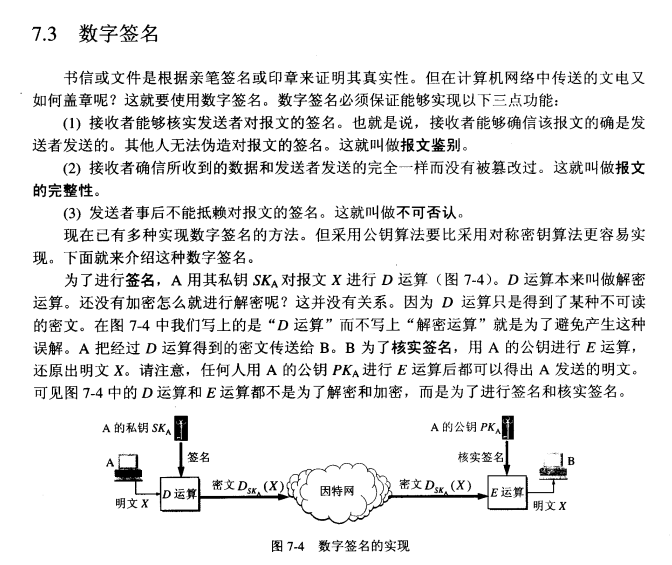
1、身份证明；

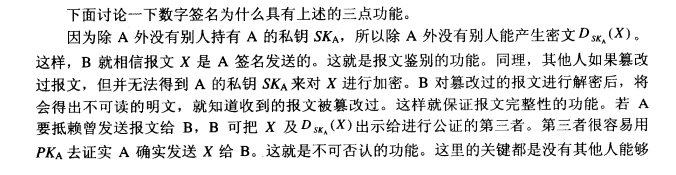
2、不可抵赖；

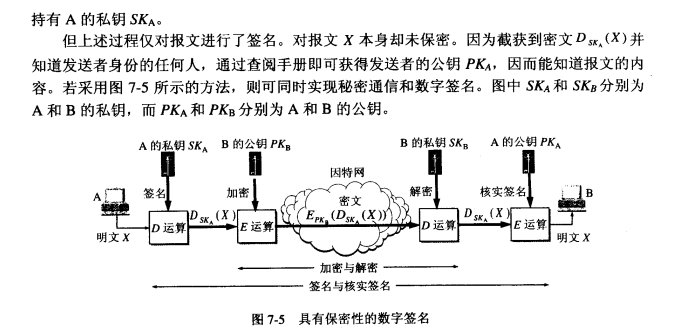
3、无法伪造；

在区块块中也是利用这三种功能，来实现区块链中交易过程节点的身份证明，交易的不可抵赖，和防止伪造的功能。

以下来自《计算机网络基础》对数字签名的原理介绍。







## 共识机制：

区块链技术中所采用的共识机制，是为了解决在各个节点之间如何就新的区块的产生达成一致。

### 拜占庭将军问题

#### 前提：

1、将军之间通信的信道可靠可信；

2、叛徒将军个数没有多到问题不可解；

#### 问题：

1、判断将军的个数，以便确定问题是否可解？

#### 目标：

在忠诚将军之间达成**正确的**进攻与否的**一致**意见。

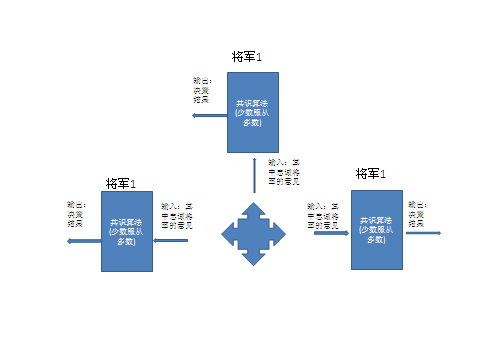
两个目标：

“一致”：这是基本目标；

“正确”：指的最终达成的一致结果应该是最大化忠诚将军利益的。（条件有利时，应该达成进攻的意见，条件不利时，应该达成撤退的意见）

如何达成“正确”：

要达成“正确”的目标，关键是每个忠诚将军能表达自己的真实意见，而不会受到叛徒意见的误导。而忠诚将军表达的真实意见是基于现有战场环境数据做出的研判，而战场环境数据里面同时包含了真伪数据，需要识别伪数据；



共识达成方法：少数服从多数；

1、每个忠诚将军表达真实意见（不会受环境伪数据）

2、所有忠诚将军接受的其他将军的指令集合（准确来说是忠诚将军的指令集合）相同；

然后根据共识算法才能产生相同的指令输出；

3、每个忠诚将军必须能采纳其他忠诚将军的指令，而不会误认为是叛徒；

归根结底，每个忠诚将军需要从输入的指令集合中识别出叛徒的指令；

# 区块链实现示例：

## 区块

### 结构：

### 操作：

创建：

就是区块结构中的各个成员的初始化；

## 区块链：

创世区块（genesis block）

### 结构：

### 操作：

#### 创建：

#### 添加新的block(入链)：

流程：

1、获取前一个block的hash，存到新block的previous hash；

2、计算新block的hash，存到新block的hash；

3、入链；

#### 区块链vlaid检查：

遍历所有的block，需要满足：重新计算的每个block hash值等于其中存储的值；每个block中存储的前一个block的hash值确实等于其前一个block的hash值；

# 智能合约（smart contract）

# 问题：

## 加密币的价值来源

区块链中的加密币的价值来源，应该是有多种。

来源0（价值起点）：

其中其最基本的或者说最小价值等于为了产生加密币所消耗的时间，电力，计算设备等资源的价值。

来源1：使用

示例：