CSE 461

Section #9:

Bitcoins

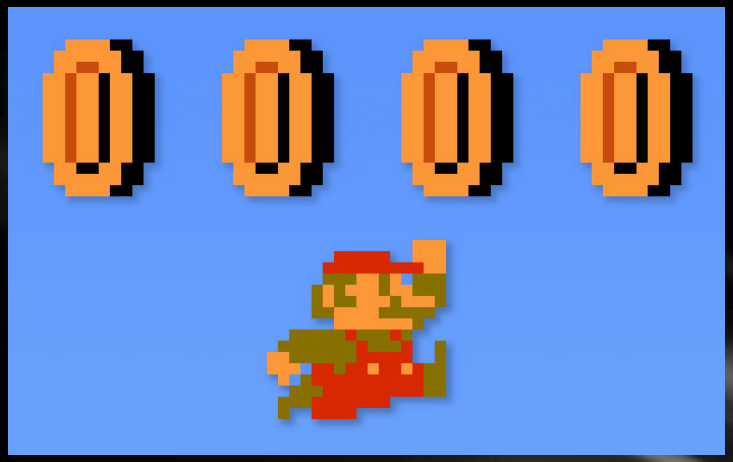
What are Bitcoins?

- Digital currency
- Unique string of bits
- Use cryptography for security and privacy
- Not tied to names: hard to trace
- Finite set of Bitcoins

Bitcoin Facts

- Created by Satoshi Nakamoto (pseudonym)
- Current Bitcoins in circulation:
  - About 12.8M BTC
- 1 BTC = ?
  - $570.17 (as of last night)
- Maximum possible number of Bitcoins:
  - ~21,000,000
- Great FAQ here:
  - https://en.bitcoin.it/wiki/FAQ

Bitcoin vocabulary



- Transaction
  - Exchange of Bitcoins
- Block
  - Record of some number of Bitcoin transactions
- Mine
  - To verify (i.e., legitimize) a block of transactions by generating a certain type of cryptographic hash
  - When a new block is generated, new Bitcoins are also generated with it
- Nonce
  - Arbitrary number used to alter hash output (more later)

Bitcoin vocabulary

- Block chain
  - A chain of blocks, each linked together with a hash of the previous block
- Genesis Block
  - Block at the beginning of a block chain which generates some number of Bitcoins
- Coinbase transaction
  - Transaction at the beginning of a block which has no inputs
  - Generates coins

Why's it hard to generate Bitcoins?

- Each block contains:
  - Version number
  - Reference to previous block (hash)
  - Merkle root (we'll explain later)
  - Creation time
  - Difficulty
  - Nonce
  - List of transactions

Why's it hard to generate Bitcoins?

- A block is not valid until a hash of the block header starts with a certain number of zeroes, determined by the "difficulty"
- E.g., if the difficulty is 3, we need 3 bytes of zeroes at the beginning of the hash:
  - 0x00000045F3B2…. Valid?
    - Yes
  - 0x00000387ECD…. Valid?
    - No
  - 0x583F23940FBA2…. Valid?
    - No

  - Find a valid hash by trying different nonces
  - Once we find a valid hash, we have a valid block, so weadd it to block chain
    - Bitcoins are generated, yay!

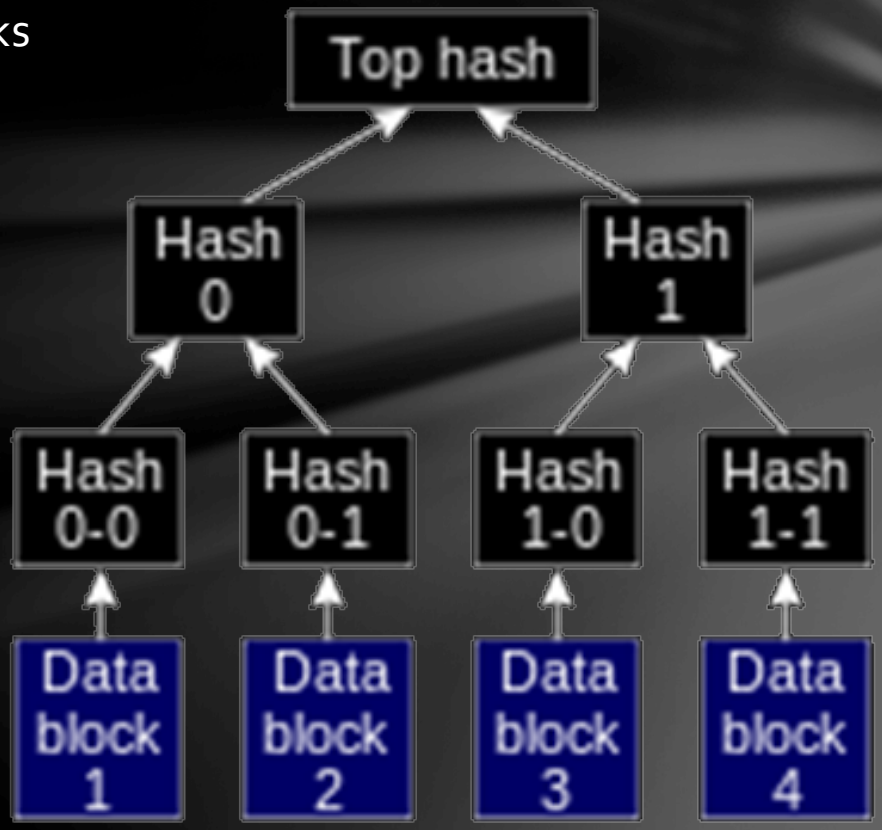
  - Generating hashes like the above is hard: why?
    - Cryptographic hashes!

Digital Signatures

- Cryptographic hashes
  - Used as "names" in Bitcoin
  - Hash a block of data, then refer to it by its hash
- Public-private key encryption
  - Talked about last lecture
- Cryptographic signatures
  - Use your private key to encrypt data
  - Anyone who knows your public key can decrypt it
  - This shows that you generated the data
- Why would signatures be useful for Bitcoin?
  - Used to prove transactions were generated by the party from whom the Bitcoins are being transferred
- Public keys are used to name transaction recipients

Merkle Trees

- Binary tree of hashes
- Leaf nodes are hashes of data blocks
- Hashes are hashed in pairs
- One hash at the top:
  - The "Merkle root"
- The Merkle root of all of the transactions in a block is included in the block header



Project 4

- Verify a set of transactions
  - Get transaction data over the network, or locally as a file
  - Throw away invalid transactions
  - Either as one block, or as many blocks (in a block chain)
  - Grouping into many blocks will generate more Bitcoins
- Keep track of Bitcoin balances for various parties listed in the transactions
- Submit code, block chain data, and ending balance list
- Competitive mining