MT Bitcoin Simulation

Syntax (Unabhängig von Zeilen):
<[*BEFEHL*]>[DATEN]

Syntax (Mehrzeilig):
<[*BEFEHL*]>[ZEILE 1]
[ZEILE 2]
[ZEILE …]

Die „[]“ nicht mitschreiben.

Auflistung aller verfügbaren Befehle:

- *hash sha256*
  - ✗ Verfügbar für Person 1, 2, 3 und 4
  - ✗ DATEN werden gehasht
- *mine hash*
  - ✗ Verfügbar für Person 1 und 2
  - ✗ ZEILE 1 gibt die Difficulty an (Ganzzahl zwischen 0 und 15)
  - ✗ ZEILE 2 wird gehasht
- *elliptic secp256k1*
  - ✗ Verfügbar für Person 1, 2, 3 und 4
  - ✗ DATEN werden mit secp256k1 umgewandelt (hexadezimaler Hash sha256)
- *pubKey to Adress*
  - ✗ Verfügbar für Person 1, 2, 3 und 4
  - ✗ DATEN werden in Adresse umgewandelt (pubKey bzw. elliptic secp256k1)
- *sign data*
  - ✗ Verfügbar für Person 1, 2, 3 und 4
  - ✗ ZEILE 1 enthält die Daten, die signiert werden sollen
  - ✗ ZEILE 2 enthält den privKey (Hexadezimaler Hash sha256)
- *verify data*
  - ✗ Verfügbar für Person 1, 2 und 3
  - ✗ ZEILE 1 enthält die Daten, die überprüft werden sollen
  - ✗ ZEILE 2 enthält die Signatur (nur Signaturen verwenden, die mit diesem Programm erzeugt wurden)
  - ✗ gibt den pubKey zurück, der zu dem privKey gehört
- *new Transaction*
  - ✗ Verfügbar für Person 1, 2, 3 und 4
  - ✗ ZEILE 1 enthält den privKey des Senders (hexadezimaler Hash sha256)
  - ✗ ZEILE 2 enthält den zu überweisenden Betrag (positive Ganzzahl)
  - ✗ ZEILE 3 enthält die Adresse des Empfängers
  - ✗ ZEILE 4 enthält den Transaktions-Fee, den der Miner bekommt (positive Ganzzahl)
- *random Transaction*
  - ✗ Verfügbar für Person 1, 2, 3 und 4
  - ✗ Generiert eine zufällige Transaktion, auf die Konten kann nicht zugegriffen werden

- *verify Blockchain*
  - ✗ Verfügbar für Person 1, 2, 3 und 4
  - ✗ Überprüft die Gültigkeit der Blockchain (Inhalt - Hash eines Blockes, Verkettung der Blöcke, Signaturen der Transaktionen)
- *crypto balance*
  - ✗ Verfügbar für Person 1, 2, 3 und 4
  - ✗ DATEN gibt die Adresse des Kontos, dessen Kontostand man wissen möchte, an
- *new Blockchain*
  - ✗ Verfügbar in der Konsole
  - ✗ erstellt eine neue Blockchain in dem Downloads-Ordner
- *mining Block*
  - ✗ Verfügbar in der Konsole
  - ✗ erstellt einen neuen Block, mit der festgelegten Difficulty
- *set Difficulty*
  - ✗ Verfügbar in der Konsole
  - ✗ DATEN enthält die neue Difficulty für das Mining der Blöcke (Ganzzahl zwischen 0 und 15)
- *create Genesis-Block*
  - ✗ Verfügbar in der Konsole
  - ✗ erstellt den Genesis-Block (dient hier ausschließlich als Beginn der Blockchain)
- *save privKey*
  - ✗ Verfügbar in der Konsole
  - ✗ ZEILE 1 enthält die Nummer der Person (1, 2, 3, oder 4)
  - ✗ ZEILE 2 enthält den neuen privKey der entsprechenden Person (hexadezimaler Hash sha256)
  - ✗ diese privKeys werden für den Mining-Reward (Person 1 und 2) benutzt und übersichtlich in der Konsole angezeigt
- *clear*
  - ✗ Verfügbar für Person 1, 2, 3 und 4 und in der Konsole
  - ✗ Löscht das jeweilige Ausgabefenster
- *clear all*
  - ✗ Verfügbar in der Konsole
  - ✗ Löscht alle Ausgabefenster
- *off*
  - ✗ Verfügbar in der Konsole
  - ✗ Leert das gesamte Konsolenfenster, auch die Anzeige der privKeys und Adressen