Project: Ethereum

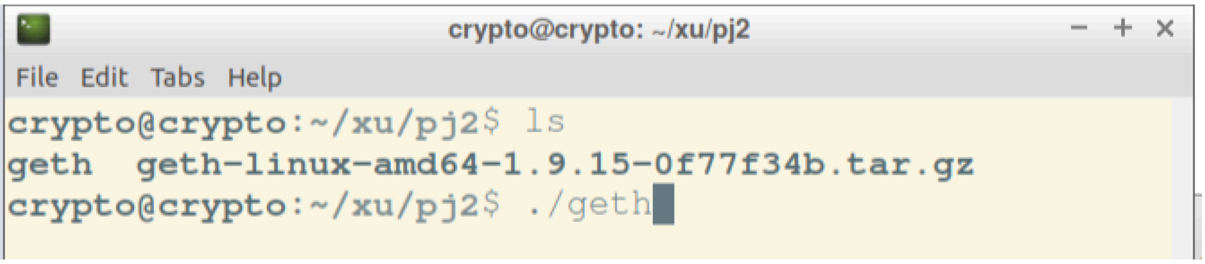
1 Project Overview

The main objective for this project is to gain first-hand experience on applications of Ethereum, a popular Blockchain platform, and to get familiar with the concepts in the Public-Key signature, hash function, key management and proof-of-work. With this project, students will learn how cryptographic algorithms are combined for applications in practice and build incentives to learn cryptographic algorithms.

The task description and expected results are given in the project description. You need to find out how to achieve that, e.g., read the given links and reading materials, search the problems and review/preview lecture slides. Please review cryptographic algorithms used in the project, identify the input and output of the algorithms, think of the features of the algorithms, figure out why the algorithms are used in that way, and what kind of security target is achieved via the algorithms.

The pre-compiled geth binary is available on Canvas. Note that the geth can be quite resource consuming. It is better to set at least a 2 core CPU and 3GB memory for the virtual machine.

Ethereum is fast evolving, you should use the executable binary file (geth 1.9.15) on Canvas. To run the binary geth, you may extract geth from the downloaded .gz file to current folder and use ./geth command to execute it. You may find out other flags by yourself.



```
crypto@crypto:~/xu/pj2$ ls
geth  geth-linux-amd64-1.9.15-0f77f34b.tar.gz
crypto@crypto:~/xu/pj2$ ./geth
```

Useful links

https://github.com/ethereum/go-ethereum
https://geth.ethereum.org/docs/interface/javascript-console
https://geth.ethereum.org/docs/rpc/server
https://web3js.readthedocs.io/

2 Project Assessment

The assessment will cover group work and individual contribution. The group project will be assessed by tutors in-class based on the task completion and questions on the tasks. 40% project marks are for the group assessment and 60% project marks are for students' individual work. The group component of the mark covers the overall project and how team members have worked collaboratively to implement their chosen overall task. During the assessment, the groups need to present the group work, such as codes and commands, and their results to tutors. Each student will be asked questions about the project task individually, and this will form the basis of individual assessment.

Up to three students per group, solo is also allowed. All the group members need to submit a project report with screenshots describing how do you work on the tasks. Students from the same group can submit the same report. Please list group members in the report.

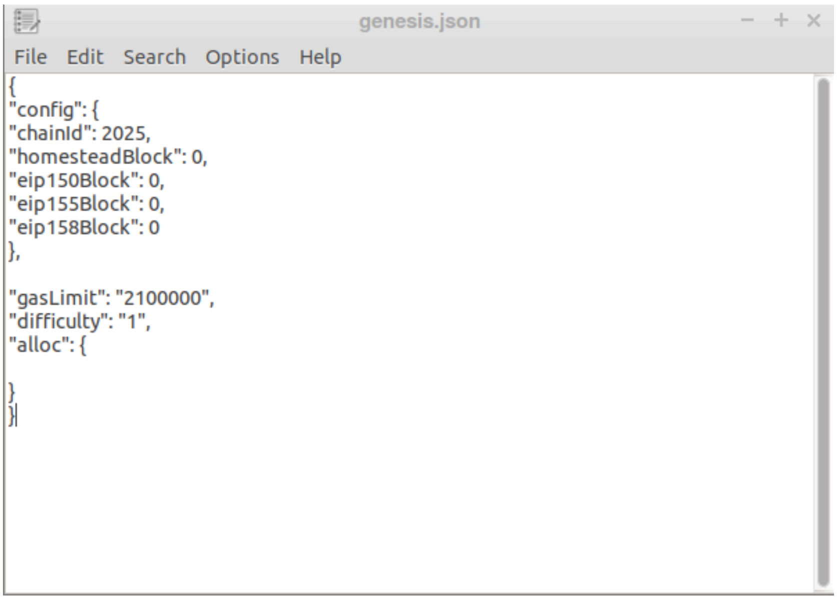
3 Project Description Part I

Run Your Private Ethereum Network with a Single Node

3.1 Task Description

1. Create a genesis.json file describing the private Ethereum network

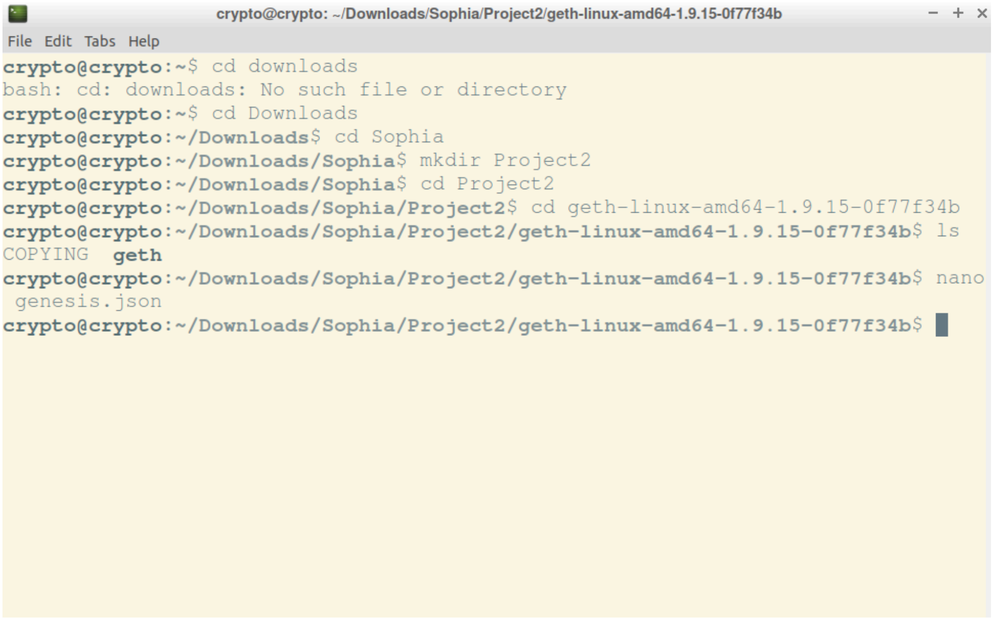
   - **Made Ethereum folder - for Ethereum network**
   - **Made json file, pasted code inside file (1st article referenced)**
     - File initialises new blockchain for Ethereum network (starting point - makes first block, next blocks derive from the first one)
     - Setting network parameters:
       - chainID - signature process, protects transactions + defence against replay attacks (retransmitting data to achieve fraudulent authentication)
       - difficulty - mining difficulty
       - gasLimit - limit cost of transactions



```
{
"config": {
"chainId": 2025,
"homesteadBlock": 0,
"eip150Block": 0,
"eip155Block": 0,
"eip158Block": 0
},

"gasLimit": "2100000",
"difficulty": "1",
"alloc": {

}
}
```

2. Use the pre-compiled **geth** binary to init the private Ethereum network

    - **pasted geth download from canvas to Project2 folder**
        - Geth - software program makes node for blockchain, written in prog language Go
            - node - Each one creates copy of entire blockchain (keeps it in sync) = validates transactions + propagates blocks
    - **changed directory (geth) to execute it**


crypto@crypto: ~/Downloads/Sophia/Project2/geth-linux-amd64-1.9.15-0f77f34b
File
Edit
Tabs
Help

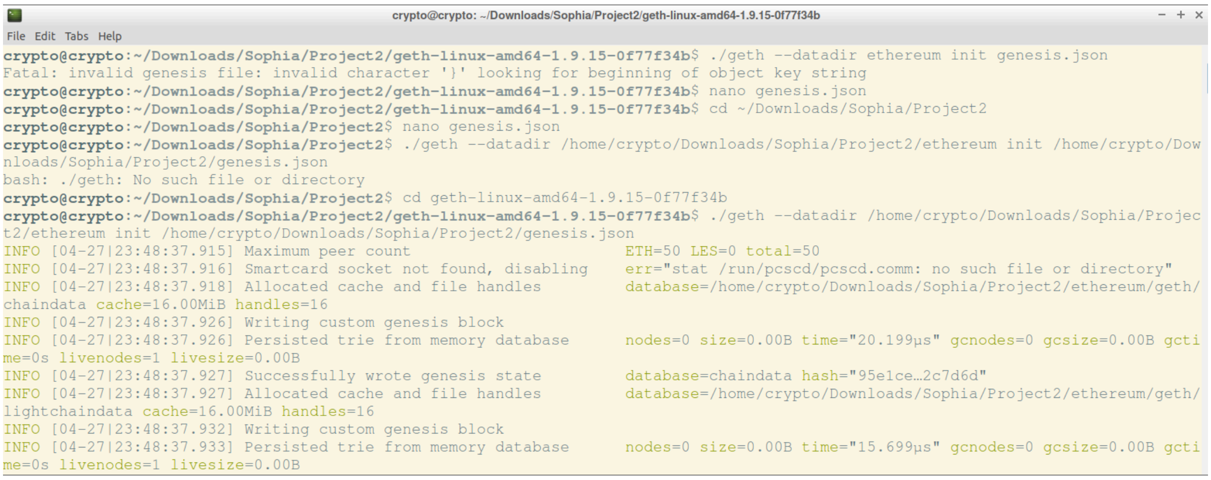
```
crypto@crypto:~$ cd downloads
bash: cd: downloads: No such file or directory
crypto@crypto:~$ cd Downloads
crypto@crypto:~/Downloads$ cd Sophia
crypto@crypto:~/Downloads/Sophia$ mkdir Project2
crypto@crypto:~/Downloads/Sophia$ cd Project2
crypto@crypto:~/Downloads/Sophia/Project2$ cd geth-linux-amd64-1.9.15-0f77f34b
crypto@crypto:~/Downloads/Sophia/Project2/geth-linux-amd64-1.9.15-0f77f34b$ ls
COPYING  geth
crypto@crypto:~/Downloads/Sophia/Project2/geth-linux-amd64-1.9.15-0f77f34b$ nano
 genesis.json
crypto@crypto:~/Downloads/Sophia/Project2/geth-linux-amd64-1.9.15-0f77f34b$ 
```

- **./geth --datadir /home/crypto/Downloads/Sophia/Project2/ethereum init /home/crypto/Downloads/Sophia/Project2/genesis.json**

    - ./ - executes geth in current directory

    - –datadir [pathname] - where blockchain data will be stored

    - innit - initialises blockchain data in genesis.json file

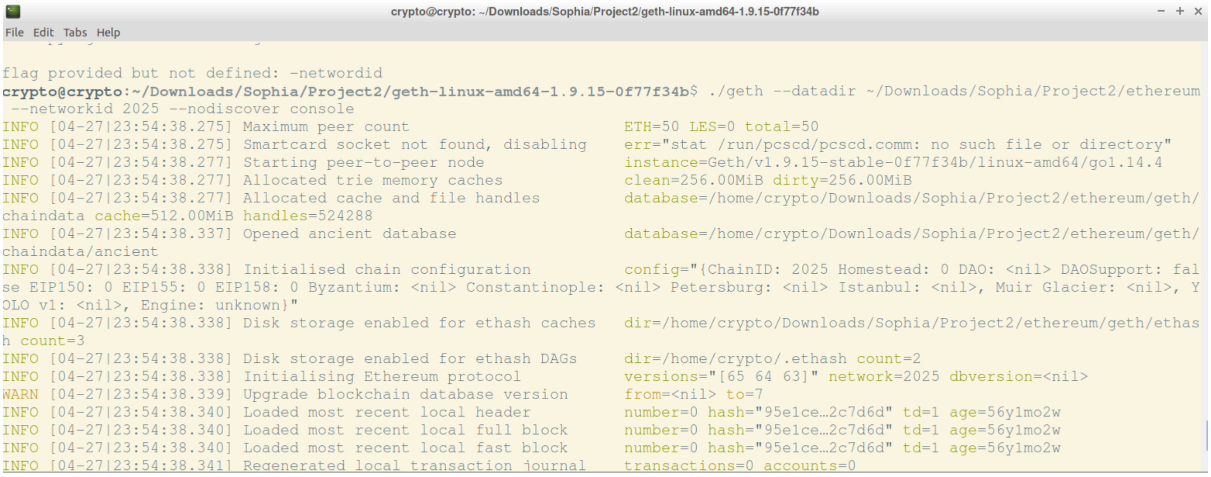

crypto@crypto: ~/Downloads/Sophia/Project2/geth-linux-amd64-1.9.15-0f77f34b
File Edit Tabs Help

```
crypto@crypto:~/Downloads/Sophia/Project2/geth-linux-amd64-1.9.15-0f77f34b$ ./geth --datadir ethereum init genesis.json
Fatal: invalid genesis file: invalid character '}' looking for beginning of object key string
crypto@crypto:~/Downloads/Sophia/Project2/geth-linux-amd64-1.9.15-0f77f34b$ nano genesis.json
crypto@crypto:~/Downloads/Sophia/Project2/geth-linux-amd64-1.9.15-0f77f34b$ cd ~/Downloads/Sophia/Project2
crypto@crypto:~/Downloads/Sophia/Project2$ nano genesis.json
crypto@crypto:~/Downloads/Sophia/Project2$ ./geth --datadir /home/crypto/Downloads/Sophia/Project2/ethereum init /home/crypto/Dow
nloads/Sophia/Project2/genesis.json
bash: ./geth: No such file or directory
crypto@crypto:~/Downloads/Sophia/Project2$ cd geth-linux-amd64-1.9.15-0f77f34b
crypto@crypto:~/Downloads/Sophia/Project2/geth-linux-amd64-1.9.15-0f77f34b$ ./geth --datadir /home/crypto/Downloads/Sophia/Projec
t2/ethereum init /home/crypto/Downloads/Sophia/Project2/genesis.json
INFO [04-27|23:48:37.915] Maximum peer count                       ETH=50 LES=0 total=50
INFO [04-27|23:48:37.916] Smartcard socket not found, disabling    err="stat /run/pcscd/pcscd.comm: no such file or directory"
INFO [04-27|23:48:37.918] Allocated cache and file handles         database=/home/crypto/Downloads/Sophia/Project2/ethereum/geth/
chaindata cache=16.00MiB handles=16
INFO [04-27|23:48:37.926] Writing custom genesis block
INFO [04-27|23:48:37.926] Persisted trie from memory database      nodes=0 size=0.00B time="20.199µs" gcnodes=0 gcsize=0.00B gcti
me=0s livenodes=1 livesize=0.00B
INFO [04-27|23:48:37.927] Successfully wrote genesis state         database=chaindata hash="95e1ce…2c7d6d"
INFO [04-27|23:48:37.927] Allocated cache and file handles         database=/home/crypto/Downloads/Sophia/Project2/ethereum/geth/
lightchaindata cache=16.00MiB handles=16
INFO [04-27|23:48:37.932] Writing custom genesis block
INFO [04-27|23:48:37.933] Persisted trie from memory database      nodes=0 size=0.00B time="15.699µs" gcnodes=0 gcsize=0.00B gcti
me=0s livenodes=1 livesize=0.00B
```

3. <u>Run your private Ethereum network and enable the geth console</u>

    - **./geth --datadir ~/Downloads/Sophia/Project2/ethereum --networkid 2025 --nodiscover console**

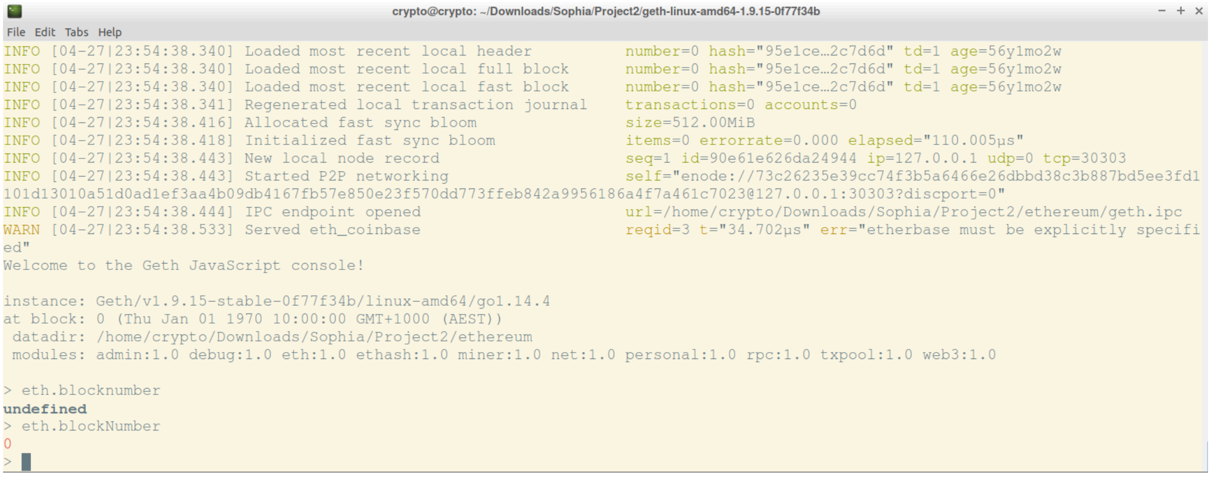
        - ./ - executes geth in current directory
        - –datadir [pathname] - where blockchain data will be stored
        - –networkid 2025 - network identifier in json file “Chainid: 2025” (differentiates other networks)
        - –nodiscover - disables discover protocol in ethereum network, preventing node from being discoverable by other peers in network (makes network private)
        - console - runs geth with Javascript console = can use query commands (mining)


crypto@crypto: ~/Downloads/Sophia/Project2/geth-linux-amd64-1.9.15-0f77f34b
File Edit Tabs Help

```
flag provided but not defined: -networdid
crypto@crypto:~/Downloads/Sophia/Project2/geth-linux-amd64-1.9.15-0f77f34b$ ./geth --datadir ~/Downloads/Sophia/Project2/ethereum
 --networkid 2025 --nodiscover console
INFO [04-27|23:54:38.275] Maximum peer count                       ETH=50 LES=0 total=50
INFO [04-27|23:54:38.275] Smartcard socket not found, disabling    err="stat /run/pcscd/pcscd.comm: no such file or directory"
INFO [04-27|23:54:38.277] Starting peer-to-peer node               instance=Geth/v1.9.15-stable-0f77f34b/linux-amd64/go1.14.4
INFO [04-27|23:54:38.277] Allocated trie memory caches             clean=256.00MiB dirty=256.00MiB
INFO [04-27|23:54:38.277] Allocated cache and file handles         database=/home/crypto/Downloads/Sophia/Project2/ethereum/geth/
chaindata cache=512.00MiB handles=524288
INFO [04-27|23:54:38.337] Opened ancient database                  database=/home/crypto/Downloads/Sophia/Project2/ethereum/geth/
chaindata/ancient
INFO [04-27|23:54:38.338] Initialised chain configuration          config="{ChainID: 2025 Homestead: 0 DAO: <nil> DAOSupport: fal
se EIP150: 0 EIP155: 0 EIP158: 0 Byzantium: <nil> Constantinople: <nil> Petersburg: <nil> Istanbul: <nil>, Muir Glacier: <nil>, Y
OLO v1: <nil>, Engine: unknown}"
INFO [04-27|23:54:38.338] Disk storage enabled for ethash caches   dir=/home/crypto/Downloads/Sophia/Project2/ethereum/geth/ethas
h count=3
INFO [04-27|23:54:38.338] Disk storage enabled for ethash DAGs     dir=/home/crypto/.ethash count=2
INFO [04-27|23:54:38.338] Initialising Ethereum protocol           versions="[65 64 63]" network=2025 dbversion=<nil>
WARN [04-27|23:54:38.339] Upgrade blockchain database version      from=<nil> to=7
INFO [04-27|23:54:38.340] Loaded most recent local header          number=0 hash="95e1ce…2c7d6d" td=1 age=56y1mo2w
INFO [04-27|23:54:38.340] Loaded most recent local full block      number=0 hash="95e1ce…2c7d6d" td=1 age=56y1mo2w
INFO [04-27|23:54:38.340] Loaded most recent local fast block      number=0 hash="95e1ce…2c7d6d" td=1 age=56y1mo2w
INFO [04-27|23:54:38.341] Regenerated local transaction journal    transactions=0 accounts=0
```

- Ethash - consensus algorithm to validate blocks = confirming they don't contain any fraudulent information
  - ○ uses Hash for verification - keccak
  - ○ POW - enables transactions to be processed without needing centralised authority
- IPC endpoint - interprocess communication = exchanging data within same network (geth within ethereum network)

4. <u>Play supported commands</u>

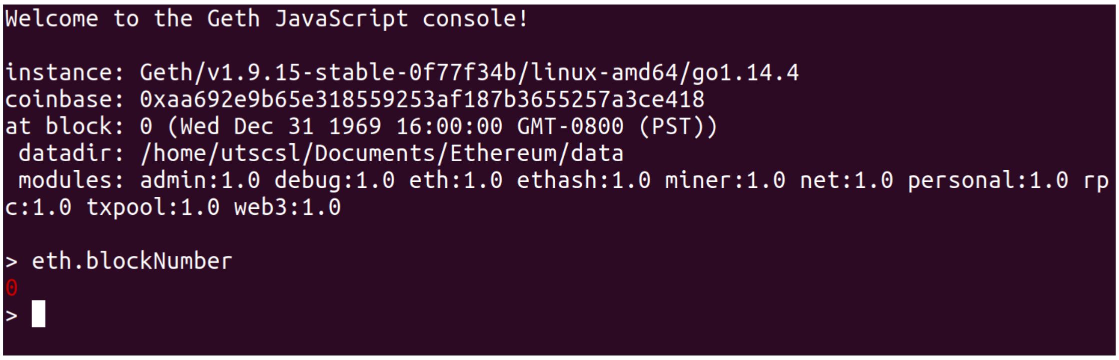

crypto@crypto: ~/Downloads/Sophia/Project2/geth-linux-amd64-1.9.15-0f77f34b
File Edit Tabs Help

```
INFO [04-27|23:54:38.340] Loaded most recent local header          number=0 hash="95e1ce…2c7d6d" td=1 age=56y1mo2w
INFO [04-27|23:54:38.340] Loaded most recent local full block      number=0 hash="95e1ce…2c7d6d" td=1 age=56y1mo2w
INFO [04-27|23:54:38.340] Loaded most recent local fast block      number=0 hash="95e1ce…2c7d6d" td=1 age=56y1mo2w
INFO [04-27|23:54:38.341] Regenerated local transaction journal    transactions=0 accounts=0
INFO [04-27|23:54:38.416] Allocated fast sync bloom                size=512.00MiB
INFO [04-27|23:54:38.418] Initialized fast sync bloom              items=0 errorrate=0.000 elapsed="110.005µs"
INFO [04-27|23:54:38.443] New local node record                    seq=1 id=90e61e626da24944 ip=127.0.0.1 udp=0 tcp=30303
INFO [04-27|23:54:38.443] Started P2P networking                   self="enode://73c26235e39cc74f3b5a6466e26dbbd38c3b887bd5ee3fd1
101d13010a51d0ad1ef3aa4b09db4167fb57e850e23f570dd773ffeb842a9956186a4f7a461c7023@127.0.0.1:30303?discport=0"
INFO [04-27|23:54:38.444] IPC endpoint opened                      url=/home/crypto/Downloads/Sophia/Project2/ethereum/geth.ipc
WARN [04-27|23:54:38.533] Served eth_coinbase                      reqid=3 t="34.702µs" err="etherbase must be explicitly specifi
ed"
Welcome to the Geth JavaScript console!

instance: Geth/v1.9.15-stable-0f77f34b/linux-amd64/go1.14.4
at block: 0 (Thu Jan 01 1970 10:00:00 GMT+1000 (AEST))
 datadir: /home/crypto/Downloads/Sophia/Project2/ethereum
 modules: admin:1.0 debug:1.0 eth:1.0 ethash:1.0 miner:1.0 net:1.0 personal:1.0 rpc:1.0 txpool:1.0 web3:1.0

> eth.blocknumber
undefined
> eth.blockNumber
0
> 
```

Hint:
The pre-compiled geth binary is available on Canvas.
You may use a small difficulty, e.g., 1, in the genesis to accelerate the block mining.
The chainID cannot be 0.
Use the "--nodiscover" option to stop your node from connecting public Ethereum network.

3.2 Expected Result

You will be able to open the geth console and input commands there.



```
Welcome to the Geth JavaScript console!

instance: Geth/v1.9.15-stable-0f77f34b/linux-amd64/go1.14.4
coinbase: 0xaa692e9b65e318559253af187b3655257a3ce418
at block: 0 (Wed Dec 31 1969 16:00:00 GMT-0800 (PST))
 datadir: /home/utscsl/Documents/Ethereum/data
 modules: admin:1.0 debug:1.0 eth:1.0 ethash:1.0 miner:1.0 net:1.0 personal:1.0 rp
c:1.0 txpool:1.0 web3:1.0

> eth.blockNumber
0
> 
```

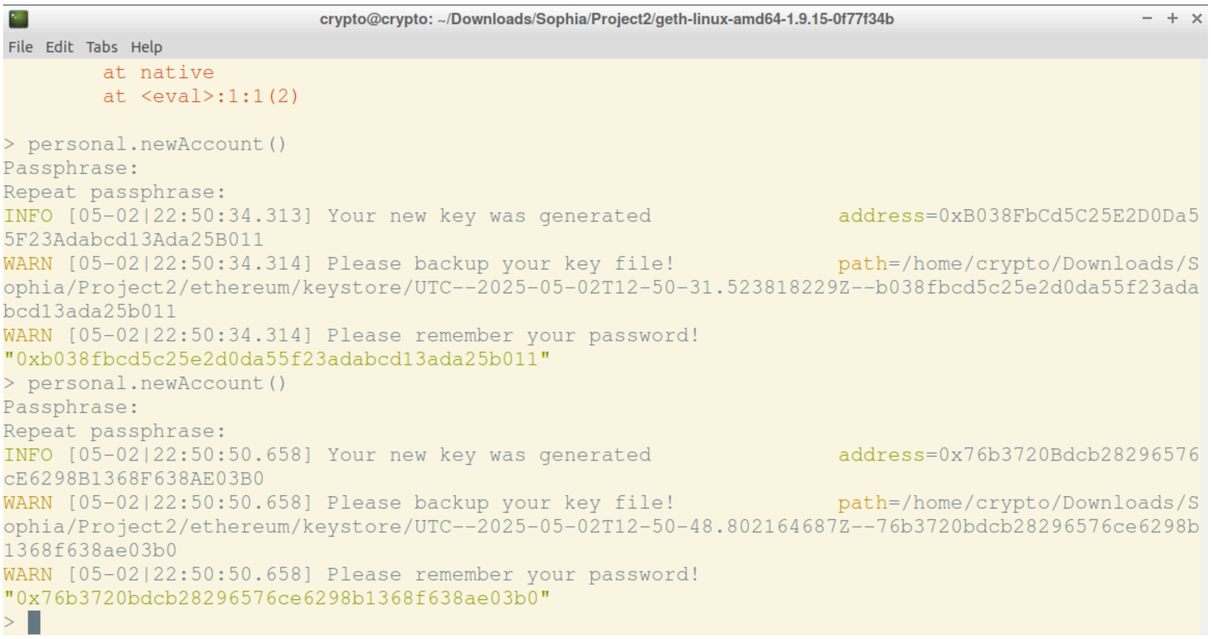
4 Project Description Part II

Create Accounts and Mining for Tokens

4.1 Task Description

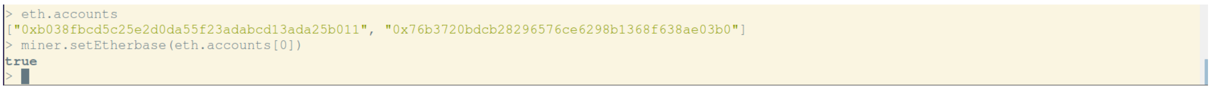
1. Create at least two accounts
    - public/private key pair - Public key cryptography
        - to securely receive/send transactions between users
        - private key = password, encrypted
        - public key = Generated by choosing point on secp256k1 elliptic curve (using private key) then adding and multiplying it
            - multiplication modulo a prime (impossible inverse – discrete logarithm problem - no trapdoor function) = protects private key
            - address - Hexadecimal numbers, Keccak 256 hash of public key, keeping last 20 bytes generate address (unique identifier - avoids digital signature forgery)

acc1	0xB038FbCd5C25E2D0Da55F23Adabcd13Ada25B011	crypto
acc2	0x76b3720Bdcb28296576cE6298B1368F638AE03B0	crypto


crypto@crypto: ~/Downloads/Sophia/Project2/geth-linux-amd64-1.9.15-0f77f34b
File Edit Tabs Help

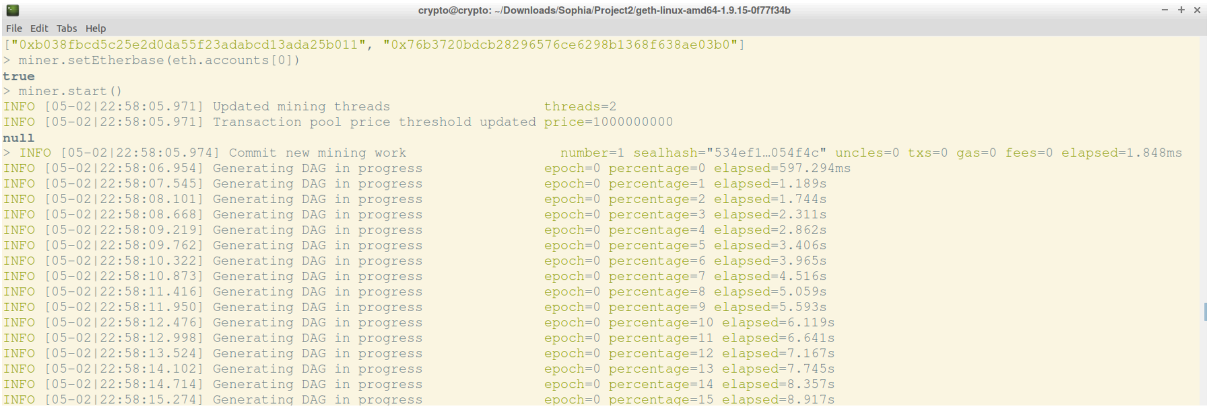
```
        at native
        at <eval>:1:1(2)

> personal.newAccount()
Passphrase:
Repeat passphrase:
INFO [05-02|22:50:34.313] Your new key was generated               address=0xB038FbCd5C25E2D0Da5
5F23Adabcd13Ada25B011
WARN [05-02|22:50:34.314] Please backup your key file!             path=/home/crypto/Downloads/S
ophia/Project2/ethereum/keystore/UTC--2025-05-02T12-50-31.523818229Z--b038fbcd5c25e2d0da55f23ada
bcd13ada25b011
WARN [05-02|22:50:34.314] Please remember your password!
"0xb038fbcd5c25e2d0da55f23adabcd13ada25b011"
> personal.newAccount()
Passphrase:
Repeat passphrase:
INFO [05-02|22:50:50.658] Your new key was generated               address=0x76b3720Bdcb28296576
cE6298B1368F638AE03B0
WARN [05-02|22:50:50.658] Please backup your key file!             path=/home/crypto/Downloads/S
ophia/Project2/ethereum/keystore/UTC--2025-05-02T12-50-48.802164687Z--76b3720bdcb28296576ce6298b
1368f638ae03b0
WARN [05-02|22:50:50.658] Please remember your password!
"0x76b3720bdcb28296576ce6298b1368f638ae03b0"
> 
```

2. <u>Set one account to be the etherbase to receive token reward from mining</u>
    - miner.setEtherbase(eth.accounts[0]) - specify first account does mining



```
> eth.accounts
["0xb038fbcd5c25e2d0da55f23adabcd13ada25b011", "0x76b3720bdcb28296576ce6298b1368f638ae03b0"]
> miner.setEtherbase(eth.accounts[0])
true
> 
```

3. <u>Mining for tokens</u>
    - Mining - secures network through creating blocks in blockchain, validates transactions, adds ether to network
    - tokens - Unique items stored in Ethereum blockchain, sold and traded through linking
        - NFT - non fungible tokens = can't be replaced

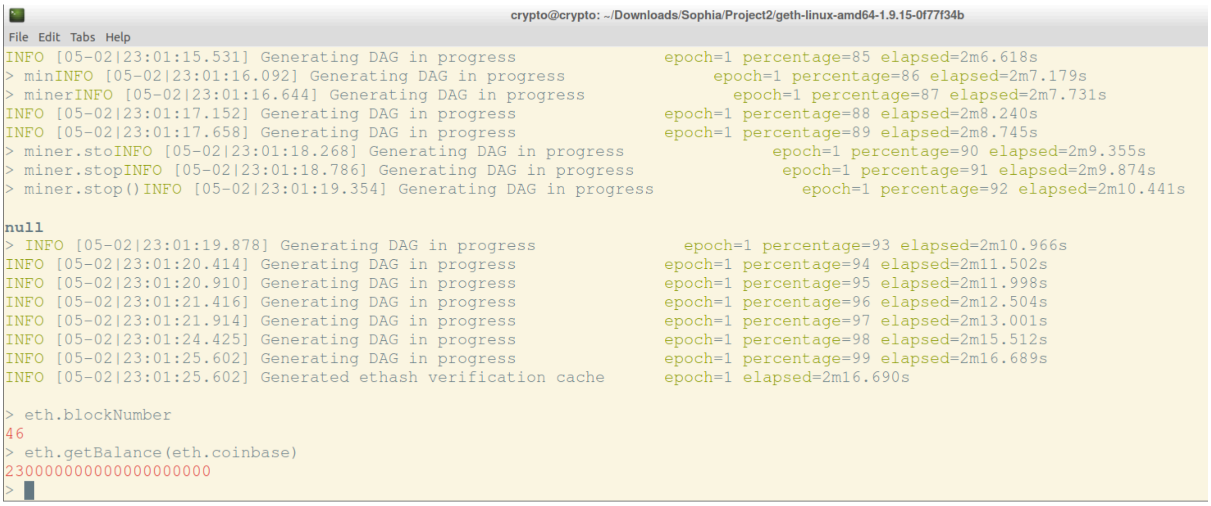


crypto@crypto: ~/Downloads/Sophia/Project2/geth-linux-amd64-1.9.15-0f77f34b
File Edit Tabs Help

```
["0xb038fbcd5c25e2d0da55f23adabcd13ada25b011", "0x76b3720bdcb28296576ce6298b1368f638ae03b0"]
> miner.setEtherbase(eth.accounts[0])
true
> miner.start()
INFO [05-02|22:58:05.971] Updated mining threads                   threads=2
INFO [05-02|22:58:05.971] Transaction pool price threshold updated price=1000000000
null
> INFO [05-02|22:58:05.974] Commit new mining work                   number=1 sealhash="534ef1…054f4c" uncles=0 txs=0 gas=0 fees=0 elapsed=1.848ms
INFO [05-02|22:58:06.954] Generating DAG in progress               epoch=0 percentage=0 elapsed=597.294ms
INFO [05-02|22:58:07.545] Generating DAG in progress               epoch=0 percentage=1 elapsed=1.189s
INFO [05-02|22:58:08.101] Generating DAG in progress               epoch=0 percentage=2 elapsed=1.744s
INFO [05-02|22:58:08.668] Generating DAG in progress               epoch=0 percentage=3 elapsed=2.311s
INFO [05-02|22:58:09.219] Generating DAG in progress               epoch=0 percentage=4 elapsed=2.862s
INFO [05-02|22:58:09.762] Generating DAG in progress               epoch=0 percentage=5 elapsed=3.406s
INFO [05-02|22:58:10.322] Generating DAG in progress               epoch=0 percentage=6 elapsed=3.965s
INFO [05-02|22:58:10.873] Generating DAG in progress               epoch=0 percentage=7 elapsed=4.516s
INFO [05-02|22:58:11.416] Generating DAG in progress               epoch=0 percentage=8 elapsed=5.059s
INFO [05-02|22:58:11.950] Generating DAG in progress               epoch=0 percentage=9 elapsed=5.593s
INFO [05-02|22:58:12.476] Generating DAG in progress               epoch=0 percentage=10 elapsed=6.119s
INFO [05-02|22:58:12.998] Generating DAG in progress               epoch=0 percentage=11 elapsed=6.641s
INFO [05-02|22:58:13.524] Generating DAG in progress               epoch=0 percentage=12 elapsed=7.167s
INFO [05-02|22:58:14.102] Generating DAG in progress               epoch=0 percentage=13 elapsed=7.745s
INFO [05-02|22:58:14.714] Generating DAG in progress               epoch=0 percentage=14 elapsed=8.357s
INFO [05-02|22:58:15.274] Generating DAG in progress               epoch=0 percentage=15 elapsed=8.917s
```

- DAG - directed acyclic graph, improves blockchain (data structure)
- epoch - time taken for 30,000 blocks to be created in blockchain

4. <u>Check the balance of the account</u>
    - 46 blockchains node has processed
    - balance of account - 230000000000000000000

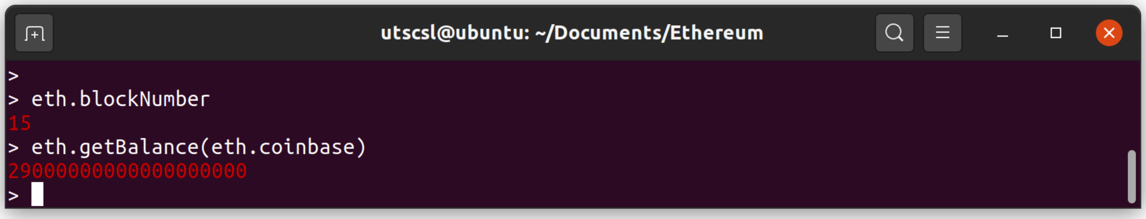


crypto@crypto: ~/Downloads/Sophia/Project2/geth-linux-amd64-1.9.15-0f77f34b
File Edit Tabs Help

```
INFO [05-02|23:01:15.531] Generating DAG in progress               epoch=1 percentage=85 elapsed=2m6.618s
> minINFO [05-02|23:01:16.092] Generating DAG in progress               epoch=1 percentage=86 elapsed=2m7.179s
> minerINFO [05-02|23:01:16.644] Generating DAG in progress               epoch=1 percentage=87 elapsed=2m7.731s
INFO [05-02|23:01:17.152] Generating DAG in progress               epoch=1 percentage=88 elapsed=2m8.240s
INFO [05-02|23:01:17.658] Generating DAG in progress               epoch=1 percentage=89 elapsed=2m8.745s
> miner.stoINFO [05-02|23:01:18.268] Generating DAG in progress               epoch=1 percentage=90 elapsed=2m9.355s
> miner.stopINFO [05-02|23:01:18.786] Generating DAG in progress               epoch=1 percentage=91 elapsed=2m9.874s
> miner.stop()INFO [05-02|23:01:19.354] Generating DAG in progress               epoch=1 percentage=92 elapsed=2m10.441s

null
> INFO [05-02|23:01:19.878] Generating DAG in progress               epoch=1 percentage=93 elapsed=2m10.966s
INFO [05-02|23:01:20.414] Generating DAG in progress               epoch=1 percentage=94 elapsed=2m11.502s
INFO [05-02|23:01:20.910] Generating DAG in progress               epoch=1 percentage=95 elapsed=2m11.998s
INFO [05-02|23:01:21.416] Generating DAG in progress               epoch=1 percentage=96 elapsed=2m12.504s
INFO [05-02|23:01:21.914] Generating DAG in progress               epoch=1 percentage=97 elapsed=2m13.001s
INFO [05-02|23:01:24.425] Generating DAG in progress               epoch=1 percentage=98 elapsed=2m15.512s
INFO [05-02|23:01:25.602] Generating DAG in progress               epoch=1 percentage=99 elapsed=2m16.689s
INFO [05-02|23:01:25.602] Generated ethash verification cache      epoch=1 elapsed=2m16.690s

> eth.blockNumber
46
> eth.getBalance(eth.coinbase)
230000000000000000000
> 
```

4.2 Expected Result

You will find the blockchain grows. Your account, which is set to be the etherbase, should receive some mining rewards. You can use the eth.getBalance() to check its balance. An example is given below.



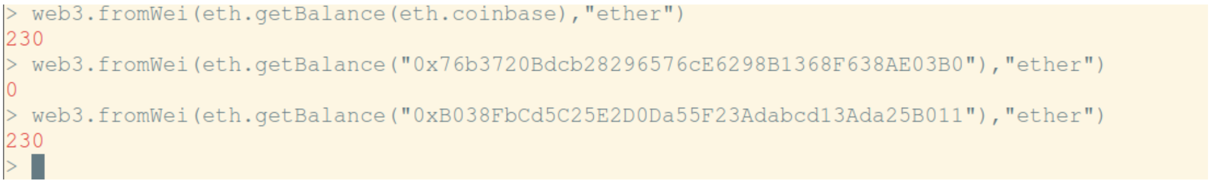
```
>
> eth.blockNumber
15
> eth.getBalance(eth.coinbase)
29000000000000000000
> 
```

5 Project Description Part III

Create transactions and mine into blocks

5.1 Task Description

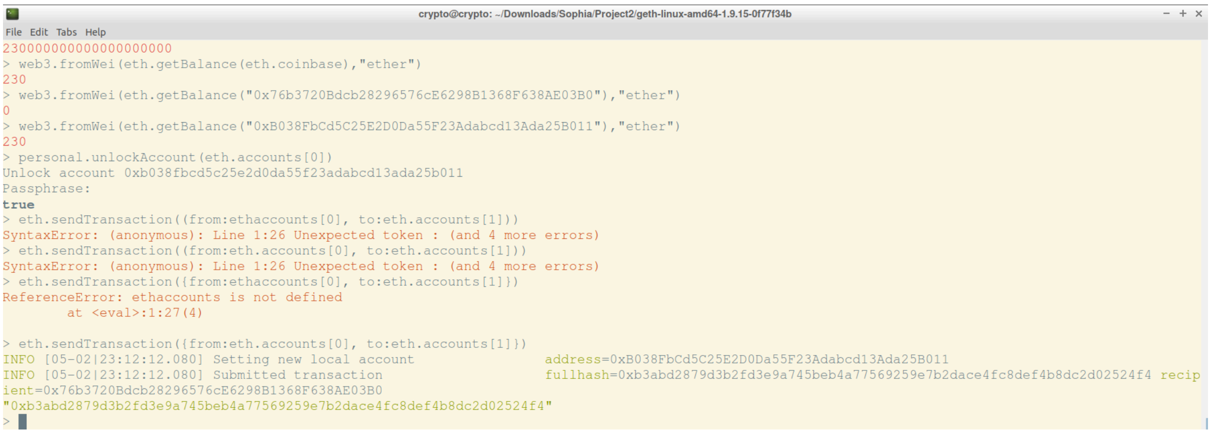
1. Stop mining and check the balance of the two accounts
   - web3 - javascript library, converts values from Wei to readable
   - taking fromWei command - wei is smallest denomination of Ether (standard readable unit)



```
> web3.fromWei(eth.getBalance(eth.coinbase),"ether")
230
> web3.fromWei(eth.getBalance("0x76b3720Bdcb28296576cE6298B1368F638AE03B0"),"ether")
0
> web3.fromWei(eth.getBalance("0xB038FbCd5C25E2D0Da55F23Adabcd13Ada25B011"),"ether")
230
> 
```

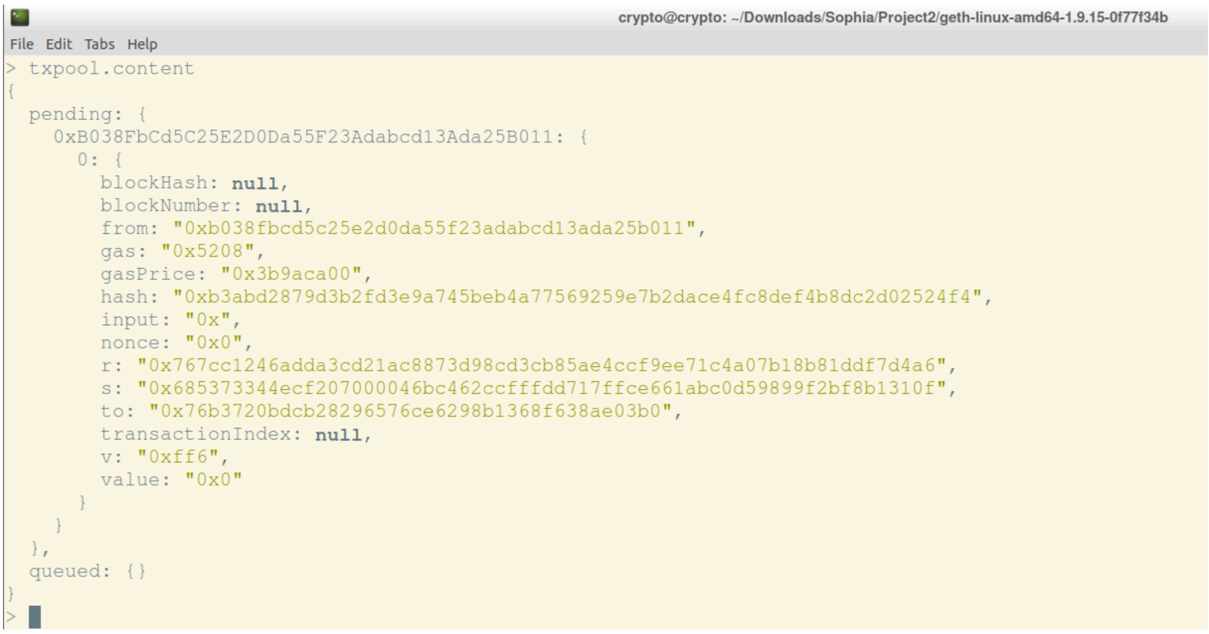
2. Create a transaction from the ether base account to another account
   - unlock acc - enables transaction to be processed (moving mining from first acc to second acc)

3. Send the transaction
   - transaction is sent to Ethereum network for validation by publicly broadcasting using geth (creates nodes) - Ethereum network is an open source blockchain-based computing platform, transparent system, uses smart contracts (automated transactions) = anyone can use without depending on centralised control + transactions can't be stopped by third party interference
   - Transactions need digital signature (full hash) to be included in blockchain
   - Digital signature = true owner of private key = control over account (verification)


crypto@crypto: ~/Downloads/Sophia/Project2/geth-linux-amd64-1.9.15-0f77f34b
File Edit Tabs Help

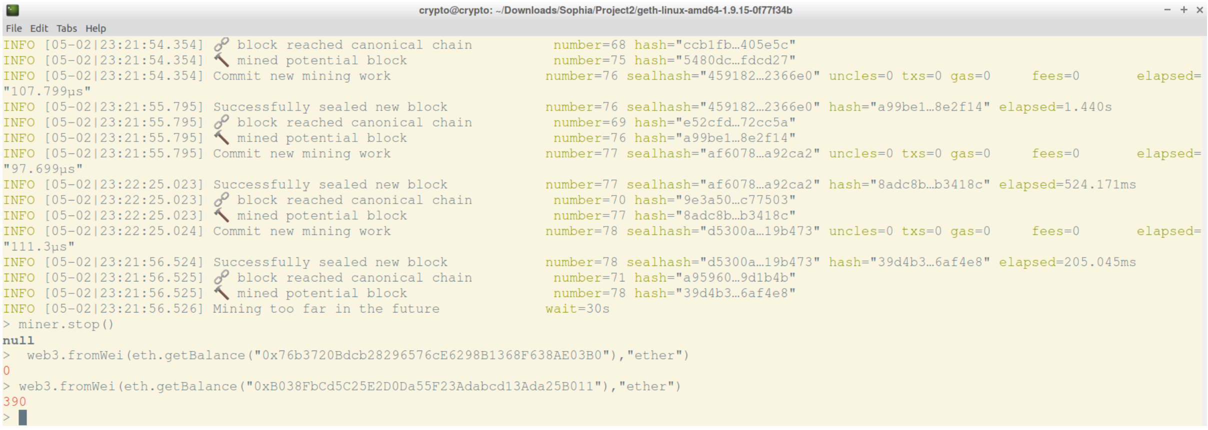
```
230000000000000000000
> web3.fromWei(eth.getBalance(eth.coinbase),"ether")
230
> web3.fromWei(eth.getBalance("0x76b3720Bdcb28296576cE6298B1368F638AE03B0"),"ether")
0
> web3.fromWei(eth.getBalance("0xB038FbCd5C25E2D0Da55F23Adabcd13Ada25B011"),"ether")
230
> personal.unlockAccount(eth.accounts[0])
Unlock account 0xb038fbcd5c25e2d0da55f23adabcd13ada25b011
Passphrase:
true
> eth.sendTransaction((from:ethaccounts[0], to:eth.accounts[1]))
SyntaxError: (anonymous): Line 1:26 Unexpected token : (and 4 more errors)
> eth.sendTransaction((from:eth.accounts[0], to:eth.accounts[1]))
SyntaxError: (anonymous): Line 1:26 Unexpected token : (and 4 more errors)
> eth.sendTransaction({from:ethaccounts[0], to:eth.accounts[1]})
ReferenceError: ethaccounts is not defined
        at <eval>:1:27(4)

> eth.sendTransaction({from:eth.accounts[0], to:eth.accounts[1]})
INFO [05-02|23:12:12.080] Setting new local account                address=0xB038FbCd5C25E2D0Da55F23Adabcd13Ada25B011
INFO [05-02|23:12:12.080] Submitted transaction                    fullhash=0xb3abd2879d3b2fd3e9a745beb4a77569259e7b2dace4fc8def4b8dc2d02524f4 recip
ient=0x76b3720Bdcb28296576cE6298B1368F638AE03B0
"0xb3abd2879d3b2fd3e9a745beb4a77569259e7b2dace4fc8def4b8dc2d02524f4"
> 
```

4. Check the transaction pool
    - verifying that transaction was added to transaction pool (stores pending transactions before being included in block)


crypto@crypto: ~/Downloads/Sophia/Project2/geth-linux-amd64-1.9.15-0f77f34b
File Edit Tabs Help

```
> txpool.content
{
  pending: {
    0xB038FbCd5C25E2D0Da55F23Adabcd13Ada25B011: {
      0: {
        blockHash: null,
        blockNumber: null,
        from: "0xb038fbcd5c25e2d0da55f23adabcd13ada25b011",
        gas: "0x5208",
        gasPrice: "0x3b9aca00",
        hash: "0xb3abd2879d3b2fd3e9a745beb4a77569259e7b2dace4fc8def4b8dc2d02524f4",
        input: "0x",
        nonce: "0x0",
        r: "0x767cc1246adda3cd21ac8873d98cd3cb85ae4ccf9ee71c4a07b18b81ddf7d4a6",
        s: "0x685373344ecf207000046bc462ccfffdd717ffce661abc0d59899f2bf8b1310f",
        to: "0x76b3720bdcb28296576ce6298b1368f638ae03b0",
        transactionIndex: null,
        v: "0xff6",
        value: "0x0"
      }
    }
  },
  queued: {}
}
> 
```

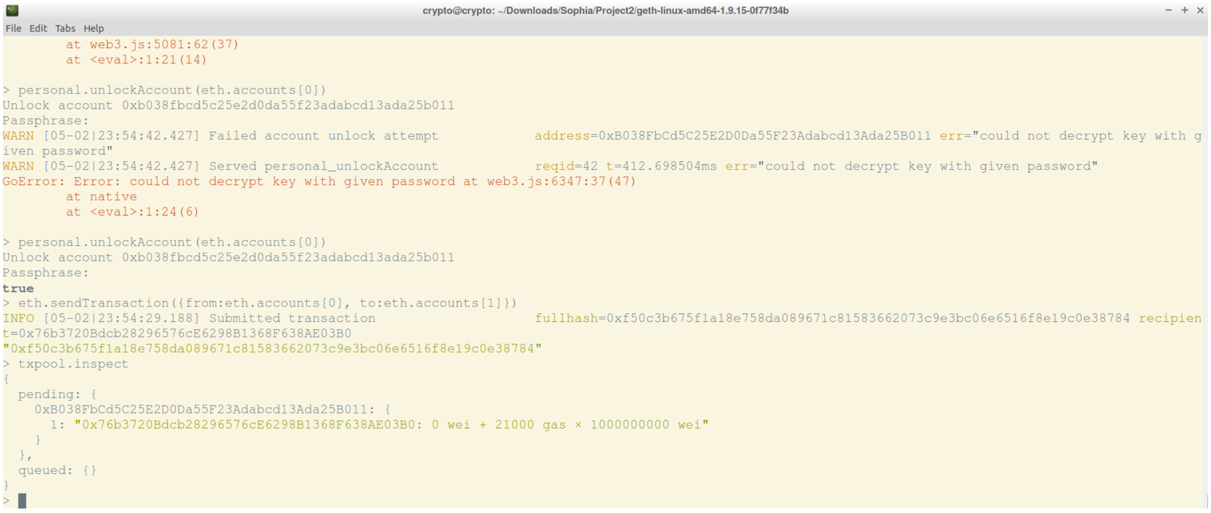
5. Start mining
6. Check the balance of the two accounts


crypto@crypto: ~/Downloads/Sophia/Project2/geth-linux-amd64-1.9.15-0f77f34b
File Edit Tabs Help

```
INFO [05-02|23:21:54.354] 🔗 block reached canonical chain          number=68 hash="ccb1fb…405e5c"
INFO [05-02|23:21:54.354] 🔨 mined potential block                  number=75 hash="5480dc…fdcd27"
INFO [05-02|23:21:54.354] Commit new mining work                   number=76 sealhash="459182…2366e0" uncles=0 txs=0 gas=0     fees=0       elapsed=
"107.799µs"
INFO [05-02|23:21:55.795] Successfully sealed new block            number=76 sealhash="459182…2366e0" hash="a99be1…8e2f14" elapsed=1.440s
INFO [05-02|23:21:55.795] 🔗 block reached canonical chain          number=69 hash="e52cfd…72cc5a"
INFO [05-02|23:21:55.795] 🔨 mined potential block                  number=76 hash="a99be1…8e2f14"
INFO [05-02|23:21:55.795] Commit new mining work                   number=77 sealhash="af6078…a92ca2" uncles=0 txs=0 gas=0     fees=0       elapsed=
"97.699µs"
INFO [05-02|23:22:25.023] Successfully sealed new block            number=77 sealhash="af6078…a92ca2" hash="8adc8b…b3418c" elapsed=524.171ms
INFO [05-02|23:22:25.023] 🔗 block reached canonical chain          number=70 hash="9e3a50…c77503"
INFO [05-02|23:22:25.023] 🔨 mined potential block                  number=77 hash="8adc8b…b3418c"
INFO [05-02|23:22:25.024] Commit new mining work                   number=78 sealhash="d5300a…19b473" uncles=0 txs=0 gas=0     fees=0       elapsed=
"111.3µs"
INFO [05-02|23:21:56.524] Successfully sealed new block            number=78 sealhash="d5300a…19b473" hash="39d4b3…6af4e8" elapsed=205.045ms
INFO [05-02|23:21:56.525] 🔗 block reached canonical chain          number=71 hash="a95960…9d1b4b"
INFO [05-02|23:21:56.525] 🔨 mined potential block                  number=78 hash="39d4b3…6af4e8"
INFO [05-02|23:21:56.526] Mining too far in the future             wait=30s
> miner.stop()
null
>  web3.fromWei(eth.getBalance("0x76b3720Bdcb28296576cE6298B1368F638AE03B0"),"ether")
0
> web3.fromWei(eth.getBalance("0xB038FbCd5C25E2D0Da55F23Adabcd13Ada25B011"),"ether")
390
> 
```

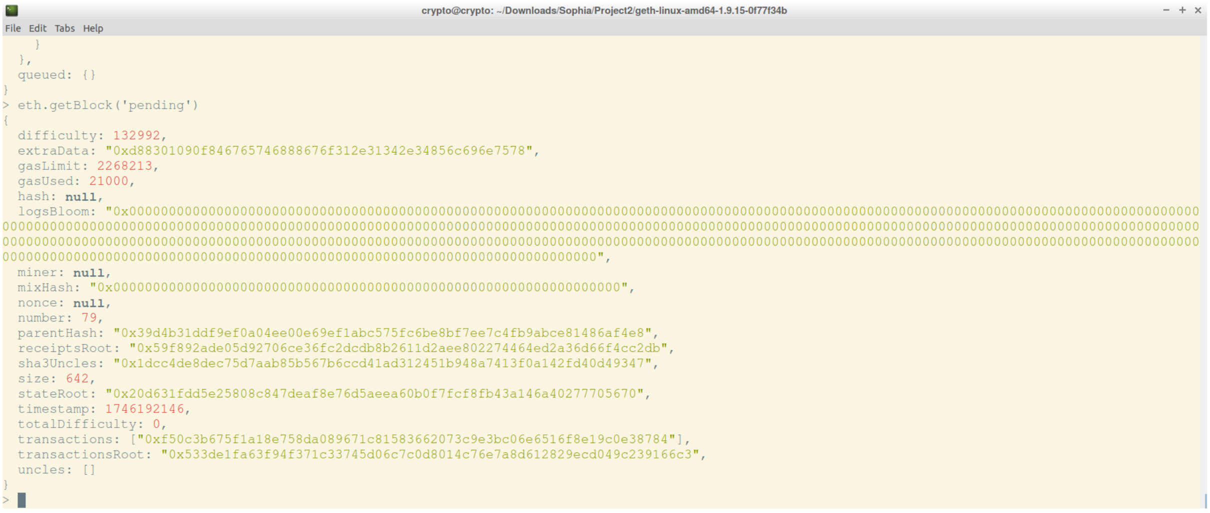
- acc 1 still has ether, not account 2 - still pending? + transaction null?

- unlocking acc, transaction
- txpool.inspect - retrieve transaction


crypto@crypto: ~/Downloads/Sophia/Project2/geth-linux-amd64-1.9.15-0f77f34b
File Edit Tabs Help

```
        at web3.js:5081:62(37)
        at <eval>:1:21(14)

> personal.unlockAccount(eth.accounts[0])
Unlock account 0xb038fbcd5c25e2d0da55f23adabcd13ada25b011
Passphrase:
WARN [05-02|23:54:42.427] Failed account unlock attempt            address=0xB038FbCd5C25E2D0Da55F23Adabcd13Ada25B011 err="could not decrypt key with g
iven password"
WARN [05-02|23:54:42.427] Served personal_unlockAccount            reqid=42 t=412.698504ms err="could not decrypt key with given password"
GoError: Error: could not decrypt key with given password at web3.js:6347:37(47)
        at native
        at <eval>:1:24(6)

> personal.unlockAccount(eth.accounts[0])
Unlock account 0xb038fbcd5c25e2d0da55f23adabcd13ada25b011
Passphrase:
true
> eth.sendTransaction({from:eth.accounts[0], to:eth.accounts[1]})
INFO [05-02|23:54:29.188] Submitted transaction                    fullhash=0xf50c3b675f1a18e758da089671c81583662073c9e3bc06e6516f8e19c0e38784 recipien
t=0x76b3720Bdcb28296576cE6298B1368F638AE03B0
"0xf50c3b675f1a18e758da089671c81583662073c9e3bc06e6516f8e19c0e38784"
> txpool.inspect
{
  pending: {
    0xB038FbCd5C25E2D0Da55F23Adabcd13Ada25B011: {
      1: "0x76b3720Bdcb28296576cE6298B1368F638AE03B0: 0 wei + 21000 gas × 1000000000 wei"
    }
  },
  queued: {}
}
> 
```

- another way of retrieving + validating transaction:

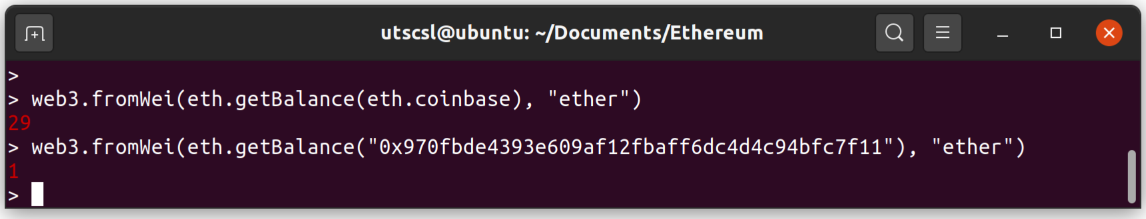


crypto@crypto: ~/Downloads/Sophia/Project2/geth-linux-amd64-1.9.15-0f77f34b
File Edit Tabs Help

```
    }
  },
  queued: {}
}
> eth.getBlock('pending')
{
  difficulty: 132992,
  extraData: "0xd88301090f846765746888676f312e31342e34856c696e7578",
  gasLimit: 2268213,
  gasUsed: 21000,
  hash: null,
  logsBloom: "0x000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000000000000000000000000000000",
  miner: null,
  mixHash: "0x0000000000000000000000000000000000000000000000000000000000000000",
  nonce: null,
  number: 79,
  parentHash: "0x39d4b31ddf9ef0a04ee00e69ef1abc575fc6be8bf7ee7c4fb9abce81486af4e8",
  receiptsRoot: "0x59f892ade05d92706ce36fc2dcdb8b2611d2aee802274464ed2a36d66f4cc2db",
  sha3Uncles: "0x1dcc4de8dec75d7aab85b567b6ccd41ad312451b948a7413f0a142fd40d49347",
  size: 642,
  stateRoot: "0x20d631fdd5e25808c847deaf8e76d5aeea60b0f7fcf8fb43a146a40277705670",
  timestamp: 1746192146,
  totalDifficulty: 0,
  transactions: ["0xf50c3b675f1a18e758da089671c81583662073c9e3bc06e6516f8e19c0e38784"],
  transactionsRoot: "0x533de1fa63f94f371c33745d06c7c0d8014c76e7a8d612829ecd049c239166c3",
  uncles: []
}
> 
```

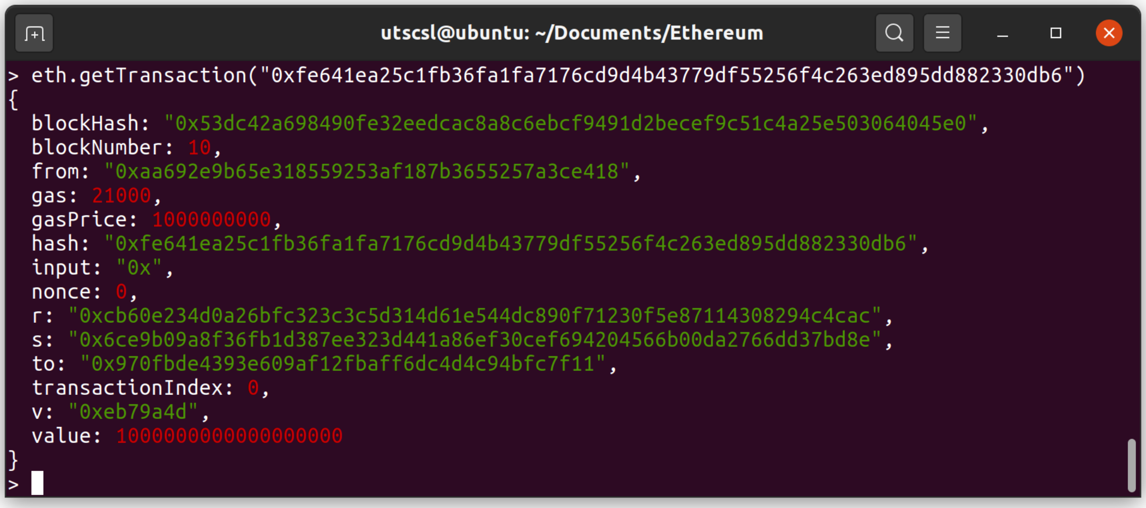
5.2 Expected Result

The receiver’s balance would be updated.



```
>
> web3.fromWei(eth.getBalance(eth.coinbase), "ether")
29
> web3.fromWei(eth.getBalance("0x970fbde4393e609af12fbaff6dc4d4c94bfc7f11"), "ether")
1
> 
```

You can also retrieve the transaction with its hash value



```
> eth.getTransaction("0xfe641ea25c1fb36fa1fa7176cd9d4b43779df55256f4c263ed895dd882330db6")
{
  blockHash: "0x53dc42a698490fe32eedcac8a8c6ebcf9491d2becef9c51c4a25e503064045e0",
  blockNumber: 10,
  from: "0xaa692e9b65e318559253af187b3655257a3ce418",
  gas: 21000,
  gasPrice: 1000000000,
  hash: "0xfe641ea25c1fb36fa1fa7176cd9d4b43779df55256f4c263ed895dd882330db6",
  input: "0x",
  nonce: 0,
  r: "0xcb60e234d0a26bfc323c3c5d314d61e544dc890f71230f5e87114308294c4cac",
  s: "0x6ce9b09a8f36fb1d387ee323d441a86ef30cef694204566b00da2766dd37bd8e",
  to: "0x970fbde4393e609af12fbaff6dc4d4c94bfc7f11",
  transactionIndex: 0,
  v: "0xeb79a4d",
  value: 1000000000000000000
}
> 
```

6 Project Description Part IV

Further Reading

Read the reading materials on Canvas which help you to get a better understanding of Ethereum and the cryptographic techniques it adopts.