September 3, 2021

CS61065 Theory and Applications of Blockchain

Assignment 2: Ethereum basics

In this assignment you will get familiar with the basics of Ethereum. You will learn how to connect to the Ethereum main network, access test networks, manage your accounts and execute transactions. You will also get started with Ethereum JavaScript API (web3.js) and do some basic operations in the test networks.

Submission Instructions

Submit the answers to the questions of Part A and Part B in the following google forms link: https://forms.gle/cQkEZSoq8BJMgJEm9

Try to make a **single submission**. In case of multiple submissions, the final one before the deadline will be considered.

Part A

Install **Geth** by following the instructions in the documentation page: https://geth.ethereum.org/docs/install-and-build/installing-geth

Create an account, and connect Geth to the test network **goerli** in **light syncmode**. Make sure that you **enable JSON-RPC over HTTP**.

Use JSON-RPC calls to answer the following questions. in the google form link provided above. Provide the (a) the answer, (b) the JSON payload used in the JSON-RPC, and (c) the response from the RPC call for each question. [Note: each question will need a single RPC call only]

A.0. Query your accounts. Write the address of the first account.

Answer:

a.
0x8af72ec4f53704fff24737f0445ddb40483eebd1

b.

```
{
	"jsonrpc":"2.0",
	"method":"eth_accounts",
	"params":[],
	"id":1
}
```

c.

```
{"jsonrpc":"2.0","id":1,"result":["0x8af72ec4f53704fff24737f0445ddb40483eebd1"]}
```

A.1. Query the current gas price in wei. Give the answer as an integer (not in hex).

A.2. Find the number of transactions in the block with hash:
“0xe019dde137734ee91ab45fc9c614cc9d636f33ab42cc875df049d45428f870f3”.
Give the answer as an integer.

A3. Query transaction receipt for the transaction with hash
“0x4210f581dda42ea2a2676fefa9edf784095a30ae2b49920e0965588d8fa78bf9”. Find out
(i) the blockNumber (integer),
(ii) blockHash,
(iii) cumulativeGasUsed (integer),
(iv) transactionIndex (integer).

A.4. Find the number of peers connected currently to your Geth client.

A.5. Get the balance of the account “0x35F18427567108F800BDC2784277B9246eED37fA”. Answer in integer (wei).

A.6. Find the hash of the transaction in block number 5417322, at index 1.

Part B

Use the Goerli faucet (https://faucet.goerli.mudit.blog/) to obtain some ethereum in your account. Then run Geth with that account unlocked using the options:
--allow-insecure-unloc
--unlock <account>
--password <path_to_file_containing_password>

Use **web3.js** to query and execute the following smart contract:

Smart Contract Address: **0x709830edf8feF92B0d879dE9ee9BdB2400BB5662**

The smart contract stores your roll number corresponding to your ethereum account address. Generate the ABI from the contract code. The code of the contract is as follows:

```
contract AddressRollMap {
    mapping(address => string) public roll;

    function update(string calldata newRoll) public {
        roll[msg.sender] = newRoll;
    }

    function get(address addr) public view returns (string memory) {
        return roll[addr];
    }

    function getmine() public view returns (string memory) {
        return roll[msg.sender];
    }
}
```

B.1. Query the roll for the address: **0x8AF72Ec4f53704FfF24737f0445ddB40483eebd1**

B.2. Input your own roll through the **update** function. **Submit your ethereum account address from which you made the transaction, and the transaction id.**