

23 August 2024

CS61065 Theory and Applications of Blockchain

Assignment 2: Ethereum Basics

Date of Submission: **August 30, 2024 EOD**

This is an individual assignment. Each student needs to submit separately. You need to login to your Google account to submit the answers. You can also use the login to edit your answer later on (within the deadline). However, if you submit multiple times from different accounts, the last submission will be considered for evaluation.

In this assignment, you will get familiar with the basics of Ethereum. You will learn how to connect to the Ethereum Test 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 in the following Google Forms link:

<https://forms.gle/erdktwiACsS7XLbLA>

Assignment Statement:

We recommend you install **Geth** by following the instructions on the documentation page and try it out. For the rest of the assignment, create an account at <https://infura.io/>, and use it to connect to the **Sepolia** test network.

Infura provides you with JSON-RPC over the HTTP interface. Use JSON-RPC calls to answer the following questions in the Google form link provided above. Provide (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].

Sample Output

Answer: 11155111 (Integer of 0xaa36a7)

JSON RPC Payload:

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

JSON Response:

```
{"jsonrpc": "2.0", "id": 1, "result": "0xaa36a7"}
```

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

Sample Answer:

- I. Answer:
- II. JSON RPC payload
- III. Response:

Q.2. Query the current latest block number (converted to decimal).

- I. Answer, II. JSON RPC payload, III. Response

Q.3. Find the balance (In Integer) of the account of a given address

0x328Ff6652cc4E79f69B165fC570e3A0F468fc903

- I. Answer, II. JSON RPC payload, III. Response

Q.4. Query the information about a block requested by block hash

"0x14aa0757da52029f170dfc5fb7fb611837303ad6fcf9949b5270dc5a53027c46"

Find Out:

- I. The Total difficulty of the chain until this block. (in Hex)
- II. The block number (In Integer)
- III. The hash of the parent block.
- IV. The root of the transaction trie of the block.
- V. JSON RPC payload
- VI. Response

Q.5. Query the number of transactions sent in the last finalized block from an address

"0x328Ff6652cc4E79f69B165fC570e3A0F468fc903"

- I. Answer (Integer), II. JSON RPC payload, III. Response

Q6. Query the information about a transaction requested by transaction hash

"0x9c155c9b20480e483c40573edca0bc7c0ffc19fc5bf05d859fbb0f0cd47799c7".

Find out

- I. value transferred in Wei(In Integer)
- II. The total amount of gas used when this transaction was executed in the block.
(in Integer)
- III. JSON RPC payload
- IV. Response

Q7. Find the number of peers currently connected (in integer) to your Geth client (in Infura) in Ethereum Sepolia Testnet.

- I. Answer (Integer), II. JSON RPC payload, III. Response

Q8. Query transaction receipt for the transaction with hash

“0xf3671a293383e03dd12639410716492c2cc600e7c7c2464aa432f24964cb585e”.

Find out

- I. the blockNumber (integer),
- II. blockHash,
- III. cumulativeGasUsed (integer),
- IV. transactionIndex (integer)
- V. JSON RPC Request
- VI. JSON Response

Q.9. Find out the number of transactions in the block with the given block: 0x4054F9

- I. Answer (Integer), II. JSON RPC payload, III. Response

Q10. Create two Accounts in Sepolia Testnet for sending a transaction. Send 0.0001Ether from Account 1 to Account 2.

- I. Public Address of Account 1
- II. Public Address of Account 2
- III. Write the hash of the transaction

