CMPE 483 Sp. Top. in CMPE Blockchain Programming
(Lab 2 – Hands on Practice)

A. Ethereum Geth Client Installation

1. Download geth from https://geth.ethereum.org/downloads/ and install it.
2. Follow the instructions in https://geth.ethereum.org/docs/developers/dapp-developer/dev-mode to start geth in development mode.
3. In another terminal window, type:

```
geth attach <IPC_LOCATION>
```

For `<IPC_LOCATION>`, substitute path of `geth.ipc` which is located data directory. You can run javascript programs in console.

B. Ethereum Geth Commands Practice

From the geth client, invoke the following commands. Example commands are given below. Please substitute your own parameters that are relevant to your address.

- `eth.getBalance(eth.accounts[0])`
- `eth.defaultAccount=eth.coinbase`
- `eth.sendTransaction({from:eth.coinbase, to: "0x3d5a6ce460da0ba161c7280d8fc6ac3d363a9124", value:web3.toWei(3, "ether"), gas:90000000, gasPrice: 200})`
- `eth.getBalance(eth.accounts[1])`
- `personal.unlockAccount(eth.accounts[0);`
- `personal.unlockAccount(eth.accounts[1], 'password', 1000000);`

```
function sleep(milliseconds) {
  var start = new Date().getTime();
  for (var i = 0; i < 1e7; i++) {
    if ((new Date().getTime() - start) > milliseconds){
      break;
    }
  }
}

function fundtransfer(fromacc,toacc,amt) {
   eth.sendTransaction({from: fromacc, to: toacc, value: web3.toWei(amt, "ether")}) ;
}

function rndtransfer(howmany,secs) {
  for(var i=0 ; i < howmany ; i++) {
     var fromacc = eth.accounts[0] ;
     var j = Math.floor((Math.random() * 5) + 1);
     var amnt = Math.floor((Math.random() * 10) + 1);
     amnt = amnt * 0.001 ;
     var toacc = eth.accounts[j] ;
     // console.log("[" + i + "] Transferring from: " + fromacc + " To: " + toacc ) ;
     fundtransfer(fromacc,toacc,amnt);
     sleep(secs*1000) ;
  }
}

rndtransfer(100,3) ;
```

C. Ethereum Smart Contract Deployment Practice

In your browser, visit the online compiler at the address: : https://remix.ethereum.org/

```
pragma solidity ^0.7.3;

// SPDX-License-Identifier: AGPL-3.0-only


contract HelloWorld  {

    string public mymessage ;

    /* constructor */
    constructor(string memory mymsg)  {
        mymessage =  mymsg ;
    }

    function message() public view returns (string memory) {
        return(mymessage);
    }

}
```

a) Compile the program and examine various fields in the online compiler window.
b) Run the above contract using the Javascript Environment.
c) Run the above contract by connecting to your local ethereum node thorugh the Web3 provider.
d) Attach a geth client to your ethereum node and run your program there using the following script.

```
var addresscontract = "0x……." ;

abicontract = [
…….
] ;

hellocontract = web3.eth.contract(abicontract).at(addresscontract);
eth.defaultAccount=eth.accounts[0] ;
personal.unlockAccount(eth.accounts[0],"password",100) ;
hellocontract.message() ;
hellocontract.message({gas:4700000}) ;
```

e) Put the above script in file “myscript.js” and run it in geth as follows:

```
loadScript("myscript.js")
```