

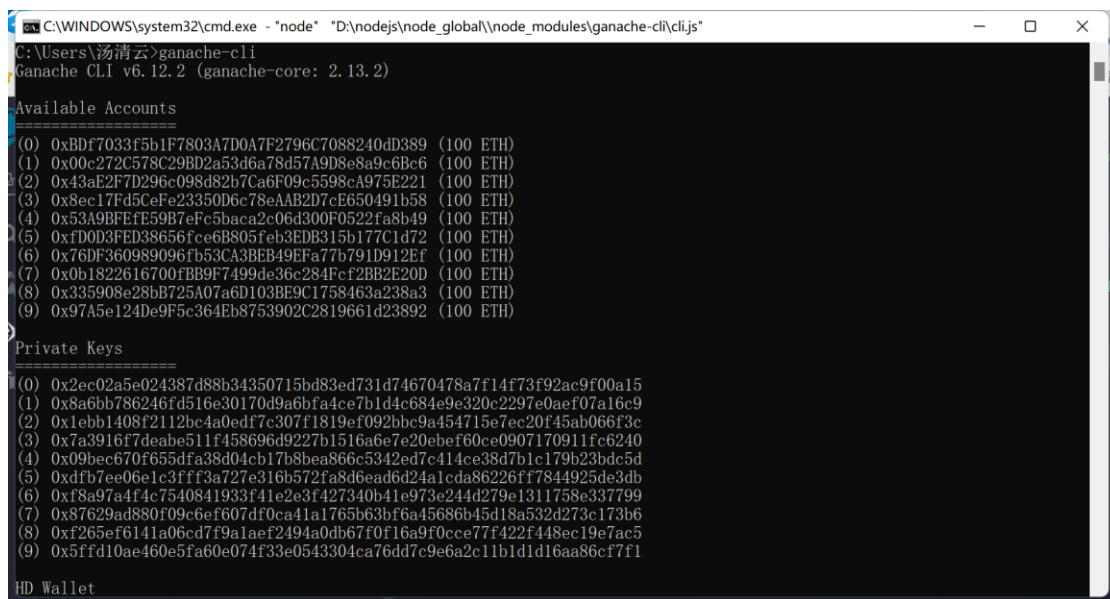
区块链第五次实验报告

2010535 赵健坤 2013536 汤清云

【注】在实验过程中我们发现原有代码前端框架同步存在严重问题，于是换用了 2022 秋斯坦福大学 CS251 的代码框架，详见 <https://cs251.stanford.edu>

【实验过程】

1. 安装 Ganache CLI，在本地命令行窗口输入 ganache-cli 来运行节点，Ctrl+C 来结束节点，其运行结果如下图：



```
C:\WINDOWS\system32\cmd.exe - "node" "D:\nodejs\node_global\node_modules\ganache-cli\cli.js"
C:\Users\汤清云>ganache-cli
Ganache CLI v6.12.2 (ganache-core: 2.13.2)

Available Accounts
=====
(0) 0xBdF7033f5b1F7803A7D0A7F2796C7088240dD389 (100 ETH)
(1) 0x00c272C578C29BD2a53d6a78d57A9D8e8a9c6Bc6 (100 ETH)
(2) 0x43aE2F7D296c098d82b7Ca6F09c5598cA975E221 (100 ETH)
(3) 0x8ec17Fd5CeFe23350D6c78eAAB2D7cE650491b58 (100 ETH)
(4) 0x53A9BFfE59B7eFe5baca2c06d300F0522fa8b49 (100 ETH)
(5) 0xfD0B3FED38656fce6B805feb3EDB315b177C1d72 (100 ETH)
(6) 0x76DF360989096fb53CA3BEB49EFa77b791D912Ef (100 ETH)
(7) 0x0b1822616700fBB9F7499de36c284Fc2BB2E20D (100 ETH)
(8) 0x335908e28bB725A07a6D103BE9C1758463a238a3 (100 ETH)
(9) 0x97A5e124De9F5c364Eb8753902C2819661d23892 (100 ETH)

Private Keys
=====
(0) 0x2ec02a5e024387d88b34350715bd83ed731d74670478a7f14f73f92ac9f00a15
(1) 0x8a6bb786246fd516e30170d9a6bfa4ce7b1d4c684e9e320c2297e0aef07a16c9
(2) 0x1ebb1408f2112bc4a0edf7c307f1819ef092bbc9a454715e7ec20f45ab066f3c
(3) 0x7a3916f7deabe511f458696d9227b1516a6e7e20ebef60ce0907170911fc6240
(4) 0x09bec670f655dfa38d04cb17b8bea866c5342ed7c414ce38d7b1c179b23bdc5d
(5) 0xdfb7ee06e1c3fff3a727e316b572fa8d6ead6d24a1cda86226ff7844925de3db
(6) 0xf8a97a4f4c7540841933f41e2e3f427340b41e973e244d279e1311758e337799
(7) 0x87629ad880f09c6ef607df0ca41a1765b63bf6a45686b45d18a532d273c173b6
(8) 0xf265ef6141a06cd7f9a1aef2494a0db67f0f16a9f0cce77f422f448ec19e7ac5
(9) 0x5ffdd10ae460e5fa60e074f33e0543304ca76dd7c9e6a2c11b1d1d16aa86cf7f1

HD Wallet
```

2. 登录 remix 网站 (<https://remix.ethereum.org>)，新建 mycontract.sol 文件，代码具体如下：

```
// Please paste your contract's solidity code here
// Note that writing a contract here WILL NOT deploy it and allow you
to access it from your client
// You should write and develop your contract in Remix and then,
before submitting, copy and paste it here

// SPDX-License-Identifier: GPL-3.0

pragma solidity >=0.7.0 <0.9.0;

/**
 * @title Splitwise
 * @dev maintain a debtbook and ensure that no cycles exist
 * @custom:dev-run-script ./scripts/deploy_with_web3.ts
```

```

*/
contract Splitwise {
    //debtbook['debtor']['creditor']保存 debtor 欠 creditor 的金额
    mapping(address => mapping(address => uint32)) debtbook;

    function lookup(address debtor, address creditor) external view
    returns (uint32 ret){
        return debtbook[debtor][creditor];
    }
    //先插入新边再消除环
    //插入边和删除环必须在一个函数内进行，否则会破坏原子性造成错误
    /**
     * @param path 欲消去的环，方向由 debtor 指向 creditor，起点与终点必须
一致
     * @param flow 环上欲消去的金额，0 代表无环
     */
    function add_IOU_chain(address creditor, uint32 amount, address[]
calldata path, uint32 flow) external{
        //为防止恶意抹去交易，必须保证数量为正
        require(amount > 0, "transaction amount must be positive!");
        //为防止前端恶意发起溢出攻击，必须防止自环
        require(creditor != msg.sender, "debtor and creditor cannot be
the same!");
        //起点与终点必须一致
        require(path[0] == path[path.length-1]);
        //STEP 1: 加入新交易
        debtbook[msg.sender][creditor] += amount;
        //此时图中存在环
        //STEP 2: 消除环
        if(flow > 0){
            //如果不是环，退出
            require(path[0]==path[path.length - 1], "path must be a
loop!");
            for(uint i = 0;i < path.length - 1;i++){
                //如果环路不完整，退出
                require(debtbook[path[i]][path[i+1]] >= flow, "loop
incomplete!");
                //更新环上的每条路径
                debtbook[path[i]][path[i+1]] -= flow;
            }
        }
    }
}

```

3. 前端代码编写如下:

```

    新增 helper function
// TODO: Add any helper functions here!
var debtBook = {};

async function addLocalIOU(debtor, creditor, amount){
    if(!(debtor in debtBook)) debtBook[debtor] = {};
    if(!(creditor in debtBook[debtor])) debtBook[debtor][creditor]
= 0;
    debtBook[debtor][creditor] += amount;
}

function resolve_cycle(path, flow) {
    if(flow == 0) return;
    //路径开始与结尾必须相同
    if(path[0]!==path[path.length-1]){
        throw new Error("The starting node and the end node must be
the same!");
    }
    for(var i = 0; i < path.length - 1; i++) {
        if(debtBook[path[i]][path[i+1]]<flow){
            throw new Error("Flow is too large!");
        }
        debtBook[path[i]][path[i+1]] -= flow;
    }
}

async function getDebtBook(){
    //clear historical data
    debtBook = {}
    var rawdata = await getAllFunctionCalls(contractAddress,
"add_IOU_chain");
    //sort by timestamp to avoid inconsistency
    rawdata.sort((a, b) => (a.t > b.t) ? 1 : -1)
    for (let index = 0; index < rawdata.length; index++) {
        const call = rawdata[index];
        addLocalIOU(call.from.toLowerCase(),
call.args[0].toLowerCase(), parseInt(call.args[1]));
        resolve_cycle(call.args[2], call.args[3]);
    }
}

async function getNeighbors(node){
    if(!(node in debtBook)) return [];

```

```

    return Object.keys(debtBook[node]);
}

function get_iou_value(debtor, creditor) {
    debtor = debtor.toLowerCase();
    creditor = creditor.toLowerCase();
    if(!(debtor in debtBook)) return 0;
    if(!(creditor in debtBook[debtor])) return 0;
    return debtBook[debtor][creditor];
}

function getMaxFlow(path){
    if(path.length<3){//自环无效
        return 0;
    }
    var maxFlow = get_iou_value(path[0], path[1]);
    for(var i = 1; i < path.length-1; i++) {
        let value = get_iou_value(path[i], path[i+1]);
        if(value < maxFlow) maxFlow = value;
    }
    return maxFlow;
}

```

getUsers 函数实现

```

async function getUsers() {
    //认定 everyone who has ever sent or received an IOU 为全部 users
    transactions = await getAllFunctionCalls(contractAddress,
    'add_IOU_chain');
    var users = new Set();
    for (let index = 0; index < transactions.length; index++) {
        const txn = transactions[index];
        users.add(txn.from.toLowerCase());
        users.add(txn.args[0]); //args 是数组型
    }
    return Array.from(users);
}

```

Gettotalowed 函数实现

```

async function getTotalOwed(user) {
    await getDebtBook();
    var sum = 0;
    if(!(user.toLowerCase() in debtBook)){
        return 0;
    }
}

```

```

    for(let amount of Object.values(debtBook[user.toLowerCase()]))
    {
        sum += amount;
    }
    return sum;
}

```

getLastActive 函数实现

```

async function getLastActive(user) {
    var rawdata = await getAllFunctionCalls(contractAddress,
"add_IOW_chain");
    var last_timestamp = null;
    for (let index = 0; index < rawdata.length; index++) {
        const call = rawdata[index];
        if((call.from.toLowerCase() === user.toLowerCase() ||
call.args[0].toLowerCase() === user.toLowerCase())
        && (last_timestamp === null || call.t >
last_timestamp)) {
            last_timestamp = call.t;
        }
    }
    return last_timestamp;
}

```

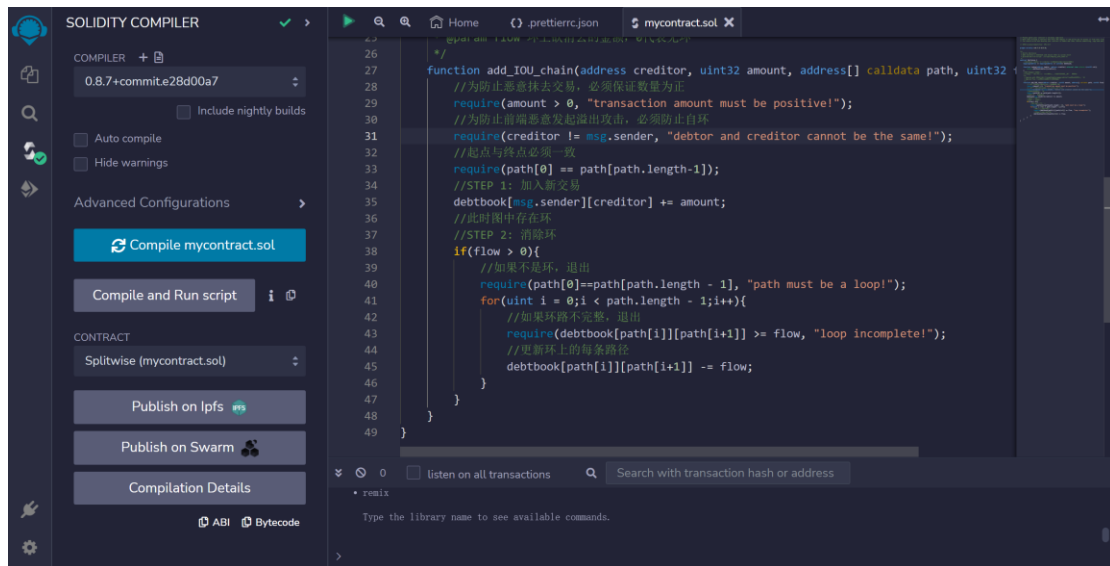
AddIOW 实现

```

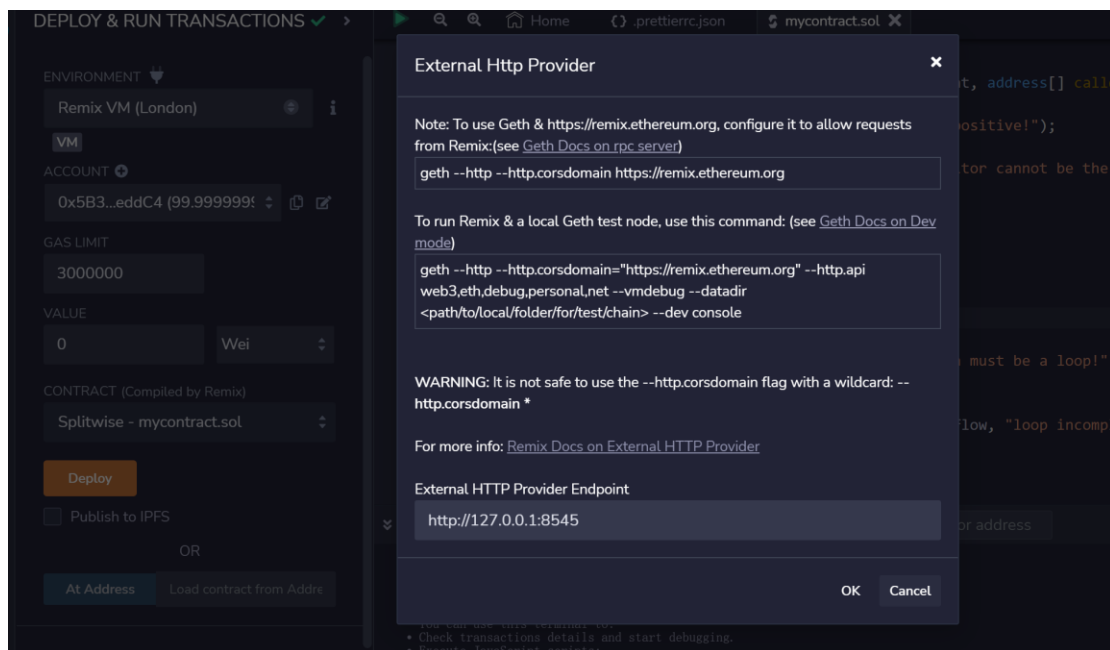
async function add_IOW(creditor, amount) {
    await getDebtBook();
    var debtor = web3.eth.defaultAccount.toLowerCase();
    addLocalIOW(debtor.toLowerCase(), creditor.toLowerCase(),
parseInt(amount));
    var cycle = await doBFS(debtor.toLowerCase(),
debtor.toLowerCase(), getNeighbors);
    var flow = await getMaxFlow(cycle);
    return
BlockchainSplitwise.methods.add_IOW_chain(creditor.toLowerCase(),
amount, cycle, flow).send({from: debtor, gas:3000000});
}

```

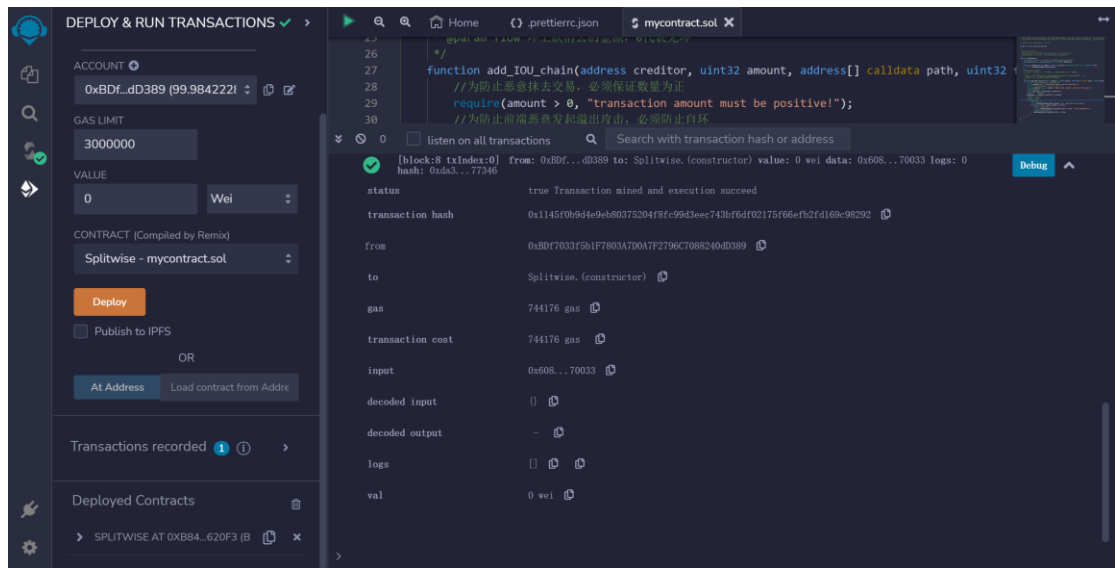
4. 在 SOLIDITY 选项卡中编译所写的文件，出现绿色的勾，说明编译成功。



5. 在 Deploy & Run 选项卡中设置环境为 External Http Provider，端点设置为 `http://localhost:8545`

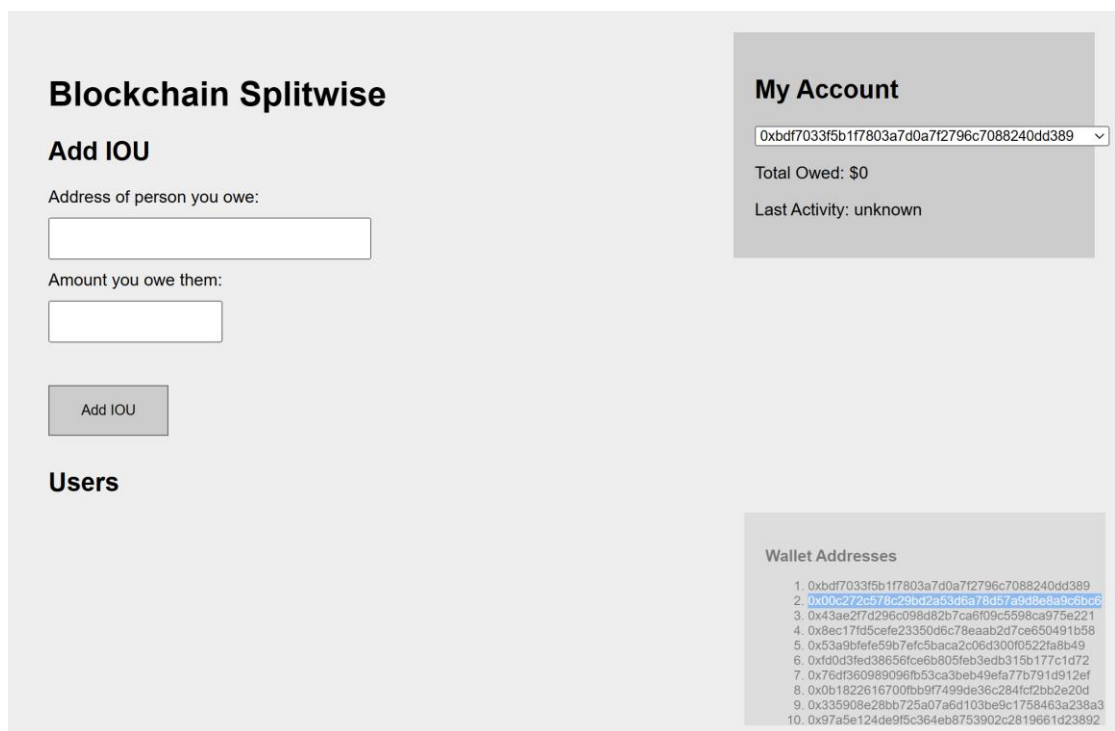


6. 运行节点并进行测试：



第一步：1 号欠 2 号 20 元：

之前：



之后：

Blockchain Splitwise

Add IOU

Address of person you owe:

Amount you owe them:

Add IOU

Users

- 0xbdf7033f5b1f7803a7d0a7f2796c7088240dd389
- 0x00c272c578c29bd2a53d6a78d57a9d8e8a9c6bc6

My Account

0xbdf7033f5b1f7803a7d0a7f2796c7088240dd389

Total Owed: \$20

Last Activity: 12/25/2022, 7:50:53 PM

Wallet Addresses

- 0xbdf7033f5b1f7803a7d0a7f2796c7088240dd389
- 0x00c272c578c29bd2a53d6a78d57a9d8e8a9c6bc6
- 0x43ae27d296c098d82b7ca6f09c5598ca975e221
- 0x8ec17fd5cfe23350d6c78eaab2d7ce650491b58
- 0x53a9bfe59b7efc5baca2c06d30f0522fa8b49
- 0xd0d3fed38656fce6b805feb3edb315b177c1d72
- 0x78d360989096b53ca30eab49efa77b7910912ef
- 0x0b1822616700fbb9f7499de36c284fc2bb2e20d
- 0x335908e28bb725a07a6d103be9c1758463a238a3
- 0x97a5e124de9f5c364eb8753902c2819661d23892

```
C:\WINDOWS\system32\cmd.exe - "node" "D:\nodejs\node_global\node_modules\ganache-cli\cli.js"
eth_blockNumber
eth_getBlockByNumber
eth_getBlockByHash
eth_getBlockByHash
eth_getBlockByHash
eth_getBlockByHash
eth_gasPrice
eth_sendTransaction

Transaction: 0x685d493758cbf34a838b5f29b197ca64644f664720a562ef71710955c49aac
Gas usage: 22336
Block Number: 5
Block Time: Sun Dec 25 2022 19:50:53 GMT+0800 (中国标准时间)

eth_getTransactionReceipt
eth_accounts
eth_accounts
eth_blockNumber
eth_blockNumber
eth_blockNumber
eth_blockNumber
eth_getBlockByNumber
eth_getBlockByNumber
eth_getBlockByNumber
eth_getBlockByNumber
eth_getTransactionReceipt
eth_getTransactionReceipt
eth_getTransactionReceipt
eth_getTransactionReceipt
eth_getBlockByNumber
```

第二步，2 号欠 3 号 20 元：

之前：

My Account

0x00c272c578c29bd2a53d6a78d57a9d8e8a9c6bc6 ▾

Total Owed: \$0

Last Activity: 12/25/2022, 7:50:53 PM

之后:

My Account

0xbdf7033f5b1f7803a7d0a7f2796c7088240dd389 ▾

Total Owed: \$20

Last Activity: 12/25/2022, 7:50:53 PM

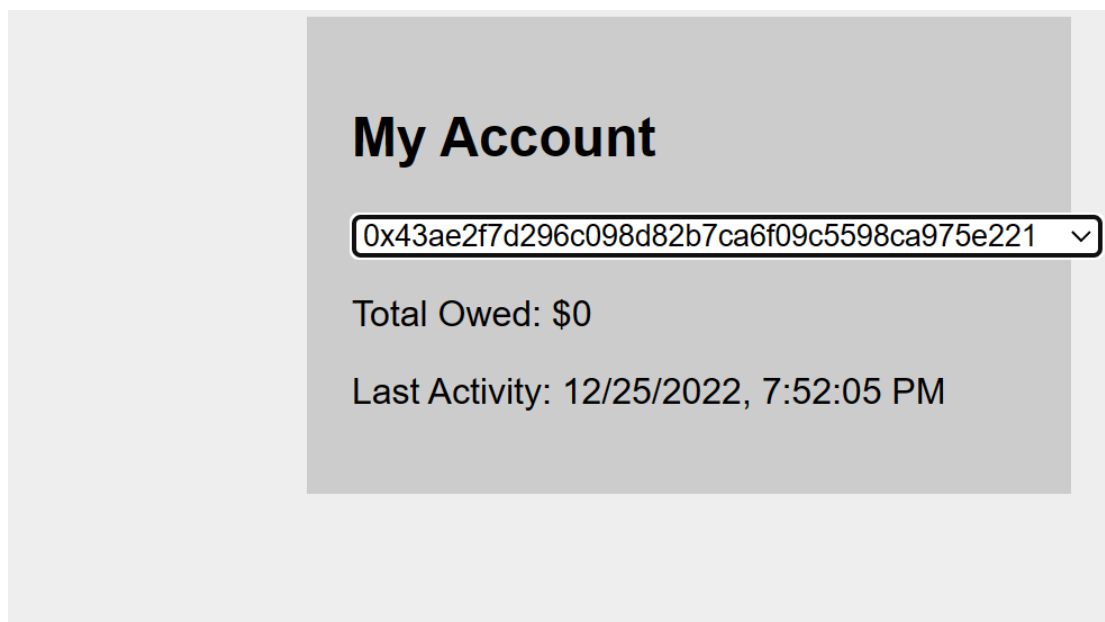
```
C:\WINDOWS\system32\cmd.exe - "node" "D:\nodejs\node_global\node_modules\ganache-cli\cli.js"
eth_blockNumber
eth_getBlockByNumber
eth_getTransactionReceipt
eth_getBlockByNumber
eth_getBlockByHash
eth_getBlockByHash
eth_getBlockByHash
eth_getBlockByHash
eth_gasPrice
eth_sendTransaction

Transaction: 0xf2db82cd75f1be24b73b1a3dcc9e66fb059da519e46cb651d9e798342f4b7ac0
Gas usage: 22336
Block Number: 6
Block Time: Sun Dec 25 2022 19:52:05 GMT+0800 (中国标准时间)

eth_getTransactionReceipt
eth_accounts
eth_accounts
eth_blockNumber
eth_blockNumber
eth_blockNumber
eth_getBlockByNumber
eth_getBlockByNumber
eth_getBlockByNumber
eth_getTransactionReceipt
eth_getTransactionReceipt
eth_getTransactionReceipt
eth_getBlockByNumber
```

第三步：3 号欠 1 号 20 元：

之前：



之后：

My Account

0xbdf7033f5b1f7803a7d0a7f2796c7088240dd389 ▾

Total Owed: \$0

Last Activity: 12/25/2022, 7:52:56 PM

```
C:\WINDOWS\system32\cmd.exe - "node" "D:\nodejs\node_global\node_modules\ganache-cli\cli.js"
eth_getBlockByHash
eth_getBlockByHash
eth_getBlockByHash
eth_getBlockByHash
eth_getBlockByHash
eth_getBlockByHash
eth_gasPrice
eth_sendTransaction

Transaction: 0x7d057f57cec69df56e75210759408f6419507395ac725b236bfe6f66db94397b
Gas usage: 23452
Block Number: 7
Block Time: Sun Dec 25 2022 19:52:56 GMT+0800 (中国标准时间)

eth_getTransactionReceipt
eth_accounts
eth_accounts
eth_blockNumber
eth_blockNumber
eth_blockNumber
eth_getBlockByNumber
eth_getBlockByNumber
eth_getBlockByNumber
eth_getTransactionReceipt
eth_getTransactionReceipt
eth_getTransactionReceipt
eth_getBlockByNumber
eth_getBlockByNumber
eth_getBlockByNumber
eth_getBlockByHash
```

交易提交后 1~3 号账户信息如下：

My Account

0xbdf7033f5b1f7803a7d0a7f2796c7088240dd389

▼

Total Owed: \$0

Last Activity: 12/25/2022, 7:52:56 PM

in Splitwise

ou owe:

m:

My Account

0x00c272c578c29bd2a53d6a78d57a9d8e8a9c6bc6

▼

Total Owed: \$0

Last Activity: 12/25/2022, 7:52:05 PM

Blockchain Splitwise

Add IOU

Address of person you owe:

Amount you owe them:

Add IOU

Users

- 0x43ae2f7d296c098d82b7ca6f09c5598ca975e221
- 0xbdf7033f5b1f7803a7d0a7f2796c7088240dd389
- 0x00c272c578c29bd2a53d6a78d57a9d8e8a9c6bc6

My Account

0x43ae2f7d296c098d82b7ca6f09c5598ca975e221

▼

Total Owed: \$0

Last Activity: 12/25/2022, 7:52:56 PM

Wallet Addresses

1. 0xbdf7033f5b1f7803a7d0a7f2796c7088240dd389

可见形成的交易环消除，实验成功。