**Федеральное государственное автономное образовательное учреждение высшего образования “Российский университет транспорта”**

**РУТ (МИИТ)   
Институт управления и цифровых технологий  
Кафедра** **“Цифровые технологии управления транспортными процессами”**

**Курсовой проект по предмету**

**Цифровые Технологии на тему:**

**“** **Реализация блокчейн (Proof-of-Work)”**

Выполнил: Костылев Е.А.

Группа: УПМ-211

Проверил: Синица Е. Ю.

**Москва 2023 г.**

Оглавление

[Цель 3](#_Toc136191289)

[Задание 3](#_Toc136191290)

[Код реализации 3](#_Toc136191291)

[main.js 3](#_Toc136191292)

[package.json 8](#_Toc136191293)

[package-lock.json 8](#_Toc136191294)

[Изменения кода 28](#_Toc136191295)

[Результаты 29](#_Toc136191296)

[Ссылка на GitHub 30](#_Toc136191297)

# Цель

Целью курсового проекта является закрепление теоретических знаний о базовых

принципах работы блокчейн технологий и майнинга.

# Задание

**В рамках курсового проекта необходимо реализовать собственный алгоритм майнинга, к примеру:**

1. Другие требования к хешу;

2. Другая математическая задача, к примеру поиск простых чисел для хеша.

# Код реализации

## main.js

'use strict';

var CryptoJS = require("crypto-js");

var express = require("express");

var bodyParser = require('body-parser');

var WebSocket = require("ws");

var http\_port = process.env.HTTP\_PORT || 3001;

var p2p\_port = process.env.P2P\_PORT || 6001;

var initialPeers = process.env.PEERS ? process.env.PEERS.split(',') : [] ;

var difficulty = 0;

class Block {

    constructor(index, previousHash, timestamp, data, hash, difficulty, nonce) {

*this*.index = index;

*this*.previousHash = previousHash.toString();

*this*.timestamp = timestamp;

*this*.data = data;

*this*.hash = hash.toString();

*this*.difficulty = difficulty;

*this*.nonce = nonce;

    }

}

var sockets = [];

var MessageType = {

QUERY\_LATEST: 0,

QUERY\_ALL : 1,

RESPONSE\_BLOCKCHAIN : 2

};

var getGenesisBlock = () = > {

    return new Block(0, "0", 1682839690, "RUT-MIIT first block",

        "8d9d5a7ff4a78042ea6737bf59c772f8ed27ef3c9b576eac1976c91aaf48d2de", 0, 0);

};

var blockchain = [getGenesisBlock()];

var initHttpServer = () = > {

    var app = express();

    app.use(bodyParser.json());

    app.get('/blocks', (req, res) = > res.send(JSON.stringify(blockchain)));

    app.post('/mineBlock', (req, res) = > {

*//var newBlock = generateNextBlock(req.body.data);*

        var newBlock = mineBlock(req.body.data);

        addBlock(newBlock);

        broadcast(responseLatestMsg());

        console.log('block added: ' + JSON.stringify(newBlock));

        res.send();

    });

    app.get('/peers', (req, res) = > {

        res.send(sockets.map(s = > s.\_socket.remoteAddress + ':' +

            s.\_socket.remotePort));

    });

    app.post('/addPeer', (req, res) = > {

        connectToPeers([req.body.peer]);

        res.send();

    });

    app.listen(http\_port, () = > console.log('Listening http on port: ' +

        http\_port));

};

var initP2PServer = () = > {

    var server = new WebSocket.Server({ port: p2p\_port });

    server.on('connection', ws = > initConnection(ws));

    console.log('listening websocket p2p port on: ' + p2p\_port);

};

var initConnection = (ws) = > {

    sockets.push(ws);

    initMessageHandler(ws);

    initErrorHandler(ws);

    write(ws, queryChainLengthMsg());

};

var initMessageHandler = (ws) = > {

    ws.on('message', (data) = > {

        var message = JSON.parse(data);

        console.log('Received message' + JSON.stringify(message));

        switch (message.type) {

        case MessageType.QUERY\_LATEST:

            write(ws, responseLatestMsg());

            break;

        case MessageType.QUERY\_ALL:

            write(ws, responseChainMsg());

            break;

        case MessageType.RESPONSE\_BLOCKCHAIN:

            handleBlockchainResponse(message);

            break;

        }

    });

};

var initErrorHandler = (ws) = > {

    var closeConnection = (ws) = > {

        console.log('connection failed to peer: ' + ws.url);

        sockets.splice(sockets.indexOf(ws), 1);

    };

    ws.on('close', () = > closeConnection(ws));

    ws.on('error', () = > closeConnection(ws));

};

var connectToPeers = (newPeers) = > {

    newPeers.forEach((peer) = > {

        var ws = new WebSocket(peer);

        ws.on('open', () = > initConnection(ws));

        ws.on('error', () = > {

            console.log('connection failed')

        });

    });

};

var handleBlockchainResponse = (message) = > {

    var receivedBlocks = JSON.parse(message.data).sort((b1, b2) = > (b1.index -

        b2.index));

    var latestBlockReceived = receivedBlocks[receivedBlocks.length - 1];

    var latestBlockHeld = getLatestBlock();

    if (latestBlockReceived.index > latestBlockHeld.index) {

        console.log('blockchain possibly behind. We got: ' +

            latestBlockHeld.index + ' Peer got: ' + latestBlockReceived.index);

        if (latestBlockHeld.hash == = latestBlockReceived.previousHash) {

            console.log("We can append the received block to our chain");

            blockchain.push(latestBlockReceived);

            broadcast(responseLatestMsg());

        }

        else if (receivedBlocks.length == = 1) {

            console.log("We have to query the chain from our peer");

            broadcast(queryAllMsg());

        }

        else {

            console.log("Received blockchain is longer than current blockchain");

            replaceChain(receivedBlocks);

        }

    }

    else {

        console.log('received blockchain is not longer than current blockchain.Do nothing');

    }

};

var generateNextBlock = (blockData) = > {

    var previousBlock = getLatestBlock();

    var nextIndex = previousBlock.index + 1;

    var nextTimestamp = new Date().getTime() / 1000;

    var nextHash = calculateHash(nextIndex, previousBlock.hash, nextTimestamp,

        blockData);

    return new Block(nextIndex, previousBlock.hash, nextTimestamp, blockData,

        nextHash);

};

var calculateHashForBlock = (block) = > {

    return calculateHash(block.index, block.previousHash, block.timestamp,

        block.data, block.nonce);

};

*//Изменил*

var calculateHash = (index, previousHash, timestamp, data, nonce) = > {

    var hash = CryptoJS.SHA256(index + previousHash + timestamp + data + nonce).toString();

    var decimalSum = 0;

    for (var i = 0; i < hash.length; i++) {

        decimalSum += parseInt(hash[i], 16);

    }

    return decimalSum.toString();

};

var addBlock = (newBlock) = > {

    if (isValidNewBlock(newBlock, getLatestBlock())) {

        blockchain.push(newBlock);

    }

};

var isValidNewBlock = (newBlock, previousBlock) = > {

    if (previousBlock.index + 1 != = newBlock.index) {

        console.log('invalid index');

        return false;

    }

    else if (previousBlock.hash != = newBlock.previousHash) {

        console.log('invalid previoushash');

        return false;

    }

    else if (calculateHashForBlock(newBlock) != = newBlock.hash) {

        console.log(typeof(newBlock.hash) + ' ' + typeof

            calculateHashForBlock(newBlock));

        console.log('invalid hash: ' + calculateHashForBlock(newBlock) + ' ' +

            newBlock.hash);

        return false;

    }

    return true;

};

var replaceChain = (newBlocks) = > {

    if (isValidChain(newBlocks) && newBlocks.length > blockchain.length) {

        console.log('Received blockchain is valid. Replacing current blockchain with received blockchain');

        blockchain = newBlocks;

        broadcast(responseLatestMsg());

    }

    else {

        console.log('Received blockchain invalid');

    }

};

var isValidChain = (blockchainToValidate) = > {

    if (JSON.stringify(blockchainToValidate[0]) != =

        JSON.stringify(getGenesisBlock())) {

        return false;

    }

    var tempBlocks = [blockchainToValidate[0]];

    for (var i = 1; i < blockchainToValidate.length; i++) {

        if (isValidNewBlock(blockchainToValidate[i], tempBlocks[i - 1])) {

            tempBlocks.push(blockchainToValidate[i]);

        }

        else {

            return false;

        }

    }

    return true;

};

var getLatestBlock = () = > blockchain[blockchain.length - 1];

var queryChainLengthMsg = () = > ({ 'type': MessageType.QUERY\_LATEST });

var queryAllMsg = () = > ({ 'type': MessageType.QUERY\_ALL });

var responseChainMsg = () = > ({

'type': MessageType.RESPONSE\_BLOCKCHAIN, 'data' : JSON.stringify(blockchain)

    });

var responseLatestMsg = () = > ({

'type': MessageType.RESPONSE\_BLOCKCHAIN,

'data' : JSON.stringify([getLatestBlock()])

    });

var write = (ws, message) = > ws.send(JSON.stringify(message));

var broadcast = (message) = > sockets.forEach(socket = > write(socket, message));

connectToPeers(initialPeers);

initHttpServer();

initP2PServer();

var mineBlock = (blockData) = > {

    var previousBlock = getLatestBlock();

    var nextIndex = previousBlock.index + 1;

    var nonce = 0;

    var nextTimestamp = new Date().getTime() / 1000;

    var nextHash = calculateHash(nextIndex, previousBlock.hash, nextTimestamp, blockData, nonce);

    while (nextHash % 37 != = 0) {

        nonce++;

        nextTimestamp = new Date().getTime() / 1000;

        nextHash = calculateHash(nextIndex, previousBlock.hash, nextTimestamp, blockData, nonce);

        console.log("\"index\": " + nextIndex + ", \"previousHash\": " + previousBlock.hash + ", \"timestamp\": " + nextTimestamp + ", \"data\": " + blockData + ", \"hash\": " + nextHash + ", \"difficulty\": " + difficulty + ", \"nonce\": " + nonce);

    }

    return new Block(nextIndex, previousBlock.hash, nextTimestamp, blockData, nextHash, difficulty, nonce);

};

## package.json

{

  "name": "blockchain",

  "version": "1.0.0",

  "description": "",

  "scripts": {

  "start": "node main.js"

  },

  "dependencies": {

  "body-parser": "^1.15.2",

  "crypto-js": "^3.1.6",

  "express": "~4.17.3",

  "ws": "^1.1.0"

},

"engines": {

"node": ">=4.3.2"

}

}

## package-lock.json

{

  "name": "blockchain",

  "version": "1.0.0",

  "lockfileVersion": 3,

  "requires": true,

  "packages": {

    "": {

      "name": "blockchain",

      "version": "1.0.0",

      "dependencies": {

        "body-parser": "^1.15.2",

        "crypto-js": "^3.1.6",

        "express": "~4.17.3",

        "ws": "^1.1.0"

      },

      "engines": {

        "node": ">=4.3.2"

      }

    },

    "node\_modules/accepts": {

      "version": "1.3.8",

      "resolved": "https://registry.npmjs.org/accepts/-/accepts-1.3.8.tgz",

      "integrity": "sha512-PYAthTa2m2VKxuvSD3DPC/Gy+U+sOA1LAuT8mkmRuvw+NACSaeXEQ+NHcVF7rONl6qcaxV3Uuemwawk+7+SJLw==",

      "dependencies": {

        "mime-types": "~2.1.34",

        "negotiator": "0.6.3"

      },

      "engines": {

        "node": ">= 0.6"

      }

    },

    "node\_modules/array-flatten": {

      "version": "1.1.1",

      "resolved": "https://registry.npmjs.org/array-flatten/-/array-flatten-1.1.1.tgz",

      "integrity": "sha512-PCVAQswWemu6UdxsDFFX/+gVeYqKAod3D3UVm91jHwynguOwAvYPhx8nNlM++NqRcK6CxxpUafjmhIdKiHibqg=="

    },

    "node\_modules/body-parser": {

      "version": "1.20.2",

      "resolved": "https://registry.npmjs.org/body-parser/-/body-parser-1.20.2.tgz",

      "integrity": "sha512-ml9pReCu3M61kGlqoTm2umSXTlRTuGTx0bfYj+uIUKKYycG5NtSbeetV3faSU6R7ajOPw0g/J1PvK4qNy7s5bA==",

      "dependencies": {

        "bytes": "3.1.2",

        "content-type": "~1.0.5",

        "debug": "2.6.9",

        "depd": "2.0.0",

        "destroy": "1.2.0",

        "http-errors": "2.0.0",

        "iconv-lite": "0.4.24",

        "on-finished": "2.4.1",

        "qs": "6.11.0",

        "raw-body": "2.5.2",

        "type-is": "~1.6.18",

        "unpipe": "1.0.0"

      },

      "engines": {

        "node": ">= 0.8",

        "npm": "1.2.8000 || >= 1.4.16"

      }

    },

    "node\_modules/bytes": {

      "version": "3.1.2",

      "resolved": "https://registry.npmjs.org/bytes/-/bytes-3.1.2.tgz",

      "integrity": "sha512-/Nf7TyzTx6S3yRJObOAV7956r8cr2+Oj8AC5dt8wSP3BQAoeX58NoHyCU8P8zGkNXStjTSi6fzO6F0pBdcYbEg==",

      "engines": {

        "node": ">= 0.8"

      }

    },

    "node\_modules/call-bind": {

      "version": "1.0.2",

      "resolved": "https://registry.npmjs.org/call-bind/-/call-bind-1.0.2.tgz",

      "integrity": "sha512-7O+FbCihrB5WGbFYesctwmTKae6rOiIzmz1icreWJ+0aA7LJfuqhEso2T9ncpcFtzMQtzXf2QGGueWJGTYsqrA==",

      "dependencies": {

        "function-bind": "^1.1.1",

        "get-intrinsic": "^1.0.2"

      },

      "funding": {

        "url": "https://github.com/sponsors/ljharb"

      }

    },

    "node\_modules/content-disposition": {

      "version": "0.5.4",

      "resolved": "https://registry.npmjs.org/content-disposition/-/content-disposition-0.5.4.tgz",

      "integrity": "sha512-FveZTNuGw04cxlAiWbzi6zTAL/lhehaWbTtgluJh4/E95DqMwTmha3KZN1aAWA8cFIhHzMZUvLevkw5Rqk+tSQ==",

      "dependencies": {

        "safe-buffer": "5.2.1"

      },

      "engines": {

        "node": ">= 0.6"

      }

    },

    "node\_modules/content-type": {

      "version": "1.0.5",

      "resolved": "https://registry.npmjs.org/content-type/-/content-type-1.0.5.tgz",

      "integrity": "sha512-nTjqfcBFEipKdXCv4YDQWCfmcLZKm81ldF0pAopTvyrFGVbcR6P/VAAd5G7N+0tTr8QqiU0tFadD6FK4NtJwOA==",

      "engines": {

        "node": ">= 0.6"

      }

    },

    "node\_modules/cookie": {

      "version": "0.4.2",

      "resolved": "https://registry.npmjs.org/cookie/-/cookie-0.4.2.tgz",

      "integrity": "sha512-aSWTXFzaKWkvHO1Ny/s+ePFpvKsPnjc551iI41v3ny/ow6tBG5Vd+FuqGNhh1LxOmVzOlGUriIlOaokOvhaStA==",

      "engines": {

        "node": ">= 0.6"

      }

    },

    "node\_modules/cookie-signature": {

      "version": "1.0.6",

      "resolved": "https://registry.npmjs.org/cookie-signature/-/cookie-signature-1.0.6.tgz",

      "integrity": "sha512-QADzlaHc8icV8I7vbaJXJwod9HWYp8uCqf1xa4OfNu1T7JVxQIrUgOWtHdNDtPiywmFbiS12VjotIXLrKM3orQ=="

    },

    "node\_modules/crypto-js": {

      "version": "3.3.0",

      "resolved": "https://registry.npmjs.org/crypto-js/-/crypto-js-3.3.0.tgz",

      "integrity": "sha512-DIT51nX0dCfKltpRiXV+/TVZq+Qq2NgF4644+K7Ttnla7zEzqc+kjJyiB96BHNyUTBxyjzRcZYpUdZa+QAqi6Q=="

    },

    "node\_modules/debug": {

      "version": "2.6.9",

      "resolved": "https://registry.npmjs.org/debug/-/debug-2.6.9.tgz",

      "integrity": "sha512-bC7ElrdJaJnPbAP+1EotYvqZsb3ecl5wi6Bfi6BJTUcNowp6cvspg0jXznRTKDjm/E7AdgFBVeAPVMNcKGsHMA==",

      "dependencies": {

        "ms": "2.0.0"

      }

    },

    "node\_modules/depd": {

      "version": "2.0.0",

      "resolved": "https://registry.npmjs.org/depd/-/depd-2.0.0.tgz",

      "integrity": "sha512-g7nH6P6dyDioJogAAGprGpCtVImJhpPk/roCzdb3fIh61/s/nPsfR6onyMwkCAR/OlC3yBC0lESvUoQEAssIrw==",

      "engines": {

        "node": ">= 0.8"

      }

    },

    "node\_modules/destroy": {

      "version": "1.2.0",

      "resolved": "https://registry.npmjs.org/destroy/-/destroy-1.2.0.tgz",

      "integrity": "sha512-2sJGJTaXIIaR1w4iJSNoN0hnMY7Gpc/n8D4qSCJw8QqFWXf7cuAgnEHxBpweaVcPevC2l3KpjYCx3NypQQgaJg==",

      "engines": {

        "node": ">= 0.8",

        "npm": "1.2.8000 || >= 1.4.16"

      }

    },

    "node\_modules/ee-first": {

      "version": "1.1.1",

      "resolved": "https://registry.npmjs.org/ee-first/-/ee-first-1.1.1.tgz",

      "integrity": "sha512-WMwm9LhRUo+WUaRN+vRuETqG89IgZphVSNkdFgeb6sS/E4OrDIN7t48CAewSHXc6C8lefD8KKfr5vY61brQlow=="

    },

    "node\_modules/encodeurl": {

      "version": "1.0.2",

      "resolved": "https://registry.npmjs.org/encodeurl/-/encodeurl-1.0.2.tgz",

      "integrity": "sha512-TPJXq8JqFaVYm2CWmPvnP2Iyo4ZSM7/QKcSmuMLDObfpH5fi7RUGmd/rTDf+rut/saiDiQEeVTNgAmJEdAOx0w==",

      "engines": {

        "node": ">= 0.8"

      }

    },

    "node\_modules/escape-html": {

      "version": "1.0.3",

      "resolved": "https://registry.npmjs.org/escape-html/-/escape-html-1.0.3.tgz",

      "integrity": "sha512-NiSupZ4OeuGwr68lGIeym/ksIZMJodUGOSCZ/FSnTxcrekbvqrgdUxlJOMpijaKZVjAJrWrGs/6Jy8OMuyj9ow=="

    },

    "node\_modules/etag": {

      "version": "1.8.1",

      "resolved": "https://registry.npmjs.org/etag/-/etag-1.8.1.tgz",

      "integrity": "sha512-aIL5Fx7mawVa300al2BnEE4iNvo1qETxLrPI/o05L7z6go7fCw1J6EQmbK4FmJ2AS7kgVF/KEZWufBfdClMcPg==",

      "engines": {

        "node": ">= 0.6"

      }

    },

    "node\_modules/express": {

      "version": "4.17.3",

      "resolved": "https://registry.npmjs.org/express/-/express-4.17.3.tgz",

      "integrity": "sha512-yuSQpz5I+Ch7gFrPCk4/c+dIBKlQUxtgwqzph132bsT6qhuzss6I8cLJQz7B3rFblzd6wtcI0ZbGltH/C4LjUg==",

      "dependencies": {

        "accepts": "~1.3.8",

        "array-flatten": "1.1.1",

        "body-parser": "1.19.2",

        "content-disposition": "0.5.4",

        "content-type": "~1.0.4",

        "cookie": "0.4.2",

        "cookie-signature": "1.0.6",

        "debug": "2.6.9",

        "depd": "~1.1.2",

        "encodeurl": "~1.0.2",

        "escape-html": "~1.0.3",

        "etag": "~1.8.1",

        "finalhandler": "~1.1.2",

        "fresh": "0.5.2",

        "merge-descriptors": "1.0.1",

        "methods": "~1.1.2",

        "on-finished": "~2.3.0",

        "parseurl": "~1.3.3",

        "path-to-regexp": "0.1.7",

        "proxy-addr": "~2.0.7",

        "qs": "6.9.7",

        "range-parser": "~1.2.1",

        "safe-buffer": "5.2.1",

        "send": "0.17.2",

        "serve-static": "1.14.2",

        "setprototypeof": "1.2.0",

        "statuses": "~1.5.0",

        "type-is": "~1.6.18",

        "utils-merge": "1.0.1",

        "vary": "~1.1.2"

      },

      "engines": {

        "node": ">= 0.10.0"

      }

    },

    "node\_modules/express/node\_modules/body-parser": {

      "version": "1.19.2",

      "resolved": "https://registry.npmjs.org/body-parser/-/body-parser-1.19.2.tgz",

      "integrity": "sha512-SAAwOxgoCKMGs9uUAUFHygfLAyaniaoun6I8mFY9pRAJL9+Kec34aU+oIjDhTycub1jozEfEwx1W1IuOYxVSFw==",

      "dependencies": {

        "bytes": "3.1.2",

        "content-type": "~1.0.4",

        "debug": "2.6.9",

        "depd": "~1.1.2",

        "http-errors": "1.8.1",

        "iconv-lite": "0.4.24",

        "on-finished": "~2.3.0",

        "qs": "6.9.7",

        "raw-body": "2.4.3",

        "type-is": "~1.6.18"

      },

      "engines": {

        "node": ">= 0.8"

      }

    },

    "node\_modules/express/node\_modules/depd": {

      "version": "1.1.2",

      "resolved": "https://registry.npmjs.org/depd/-/depd-1.1.2.tgz",

      "integrity": "sha512-7emPTl6Dpo6JRXOXjLRxck+FlLRX5847cLKEn00PLAgc3g2hTZZgr+e4c2v6QpSmLeFP3n5yUo7ft6avBK/5jQ==",

      "engines": {

        "node": ">= 0.6"

      }

    },

    "node\_modules/express/node\_modules/http-errors": {

      "version": "1.8.1",

      "resolved": "https://registry.npmjs.org/http-errors/-/http-errors-1.8.1.tgz",

      "integrity": "sha512-Kpk9Sm7NmI+RHhnj6OIWDI1d6fIoFAtFt9RLaTMRlg/8w49juAStsrBgp0Dp4OdxdVbRIeKhtCUvoi/RuAhO4g==",

      "dependencies": {

        "depd": "~1.1.2",

        "inherits": "2.0.4",

        "setprototypeof": "1.2.0",

        "statuses": ">= 1.5.0 < 2",

        "toidentifier": "1.0.1"

      },

      "engines": {

        "node": ">= 0.6"

      }

    },

    "node\_modules/express/node\_modules/on-finished": {

      "version": "2.3.0",

      "resolved": "https://registry.npmjs.org/on-finished/-/on-finished-2.3.0.tgz",

      "integrity": "sha512-ikqdkGAAyf/X/gPhXGvfgAytDZtDbr+bkNUJ0N9h5MI/dmdgCs3l6hoHrcUv41sRKew3jIwrp4qQDXiK99Utww==",

      "dependencies": {

        "ee-first": "1.1.1"

      },

      "engines": {

        "node": ">= 0.8"

      }

    },

    "node\_modules/express/node\_modules/qs": {

      "version": "6.9.7",

      "resolved": "https://registry.npmjs.org/qs/-/qs-6.9.7.tgz",

      "integrity": "sha512-IhMFgUmuNpyRfxA90umL7ByLlgRXu6tIfKPpF5TmcfRLlLCckfP/g3IQmju6jjpu+Hh8rA+2p6A27ZSPOOHdKw==",

      "engines": {

        "node": ">=0.6"

      },

      "funding": {

        "url": "https://github.com/sponsors/ljharb"

      }

    },

    "node\_modules/express/node\_modules/raw-body": {

      "version": "2.4.3",

      "resolved": "https://registry.npmjs.org/raw-body/-/raw-body-2.4.3.tgz",

      "integrity": "sha512-UlTNLIcu0uzb4D2f4WltY6cVjLi+/jEN4lgEUj3E04tpMDpUlkBo/eSn6zou9hum2VMNpCCUone0O0WeJim07g==",

      "dependencies": {

        "bytes": "3.1.2",

        "http-errors": "1.8.1",

        "iconv-lite": "0.4.24",

        "unpipe": "1.0.0"

      },

      "engines": {

        "node": ">= 0.8"

      }

    },

    "node\_modules/finalhandler": {

      "version": "1.1.2",

      "resolved": "https://registry.npmjs.org/finalhandler/-/finalhandler-1.1.2.tgz",

      "integrity": "sha512-aAWcW57uxVNrQZqFXjITpW3sIUQmHGG3qSb9mUah9MgMC4NeWhNOlNjXEYq3HjRAvL6arUviZGGJsBg6z0zsWA==",

      "dependencies": {

        "debug": "2.6.9",

        "encodeurl": "~1.0.2",

        "escape-html": "~1.0.3",

        "on-finished": "~2.3.0",

        "parseurl": "~1.3.3",

        "statuses": "~1.5.0",

        "unpipe": "~1.0.0"

      },

      "engines": {

        "node": ">= 0.8"

      }

    },

    "node\_modules/finalhandler/node\_modules/on-finished": {

      "version": "2.3.0",

      "resolved": "https://registry.npmjs.org/on-finished/-/on-finished-2.3.0.tgz",

      "integrity": "sha512-ikqdkGAAyf/X/gPhXGvfgAytDZtDbr+bkNUJ0N9h5MI/dmdgCs3l6hoHrcUv41sRKew3jIwrp4qQDXiK99Utww==",

      "dependencies": {

        "ee-first": "1.1.1"

      },

      "engines": {

        "node": ">= 0.8"

      }

    },

    "node\_modules/forwarded": {

      "version": "0.2.0",

      "resolved": "https://registry.npmjs.org/forwarded/-/forwarded-0.2.0.tgz",

      "integrity": "sha512-buRG0fpBtRHSTCOASe6hD258tEubFoRLb4ZNA6NxMVHNw2gOcwHo9wyablzMzOA5z9xA9L1KNjk/Nt6MT9aYow==",

      "engines": {

        "node": ">= 0.6"

      }

    },

    "node\_modules/fresh": {

      "version": "0.5.2",

      "resolved": "https://registry.npmjs.org/fresh/-/fresh-0.5.2.tgz",

      "integrity": "sha512-zJ2mQYM18rEFOudeV4GShTGIQ7RbzA7ozbU9I/XBpm7kqgMywgmylMwXHxZJmkVoYkna9d2pVXVXPdYTP9ej8Q==",

      "engines": {

        "node": ">= 0.6"

      }

    },

    "node\_modules/function-bind": {

      "version": "1.1.1",

      "resolved": "https://registry.npmjs.org/function-bind/-/function-bind-1.1.1.tgz",

      "integrity": "sha512-yIovAzMX49sF8Yl58fSCWJ5svSLuaibPxXQJFLmBObTuCr0Mf1KiPopGM9NiFjiYBCbfaa2Fh6breQ6ANVTI0A=="

    },

    "node\_modules/get-intrinsic": {

      "version": "1.2.1",

      "resolved": "https://registry.npmjs.org/get-intrinsic/-/get-intrinsic-1.2.1.tgz",

      "integrity": "sha512-2DcsyfABl+gVHEfCOaTrWgyt+tb6MSEGmKq+kI5HwLbIYgjgmMcV8KQ41uaKz1xxUcn9tJtgFbQUEVcEbd0FYw==",

      "dependencies": {

        "function-bind": "^1.1.1",

        "has": "^1.0.3",

        "has-proto": "^1.0.1",

        "has-symbols": "^1.0.3"

      },

      "funding": {

        "url": "https://github.com/sponsors/ljharb"

      }

    },

    "node\_modules/has": {

      "version": "1.0.3",

      "resolved": "https://registry.npmjs.org/has/-/has-1.0.3.tgz",

      "integrity": "sha512-f2dvO0VU6Oej7RkWJGrehjbzMAjFp5/VKPp5tTpWIV4JHHZK1/BxbFRtf/siA2SWTe09caDmVtYYzWEIbBS4zw==",

      "dependencies": {

        "function-bind": "^1.1.1"

      },

      "engines": {

        "node": ">= 0.4.0"

      }

    },

    "node\_modules/has-proto": {

      "version": "1.0.1",

      "resolved": "https://registry.npmjs.org/has-proto/-/has-proto-1.0.1.tgz",

      "integrity": "sha512-7qE+iP+O+bgF9clE5+UoBFzE65mlBiVj3tKCrlNQ0Ogwm0BjpT/gK4SlLYDMybDh5I3TCTKnPPa0oMG7JDYrhg==",

      "engines": {

        "node": ">= 0.4"

      },

      "funding": {

        "url": "https://github.com/sponsors/ljharb"

      }

    },

    "node\_modules/has-symbols": {

      "version": "1.0.3",

      "resolved": "https://registry.npmjs.org/has-symbols/-/has-symbols-1.0.3.tgz",

      "integrity": "sha512-l3LCuF6MgDNwTDKkdYGEihYjt5pRPbEg46rtlmnSPlUbgmB8LOIrKJbYYFBSbnPaJexMKtiPO8hmeRjRz2Td+A==",

      "engines": {

        "node": ">= 0.4"

      },

      "funding": {

        "url": "https://github.com/sponsors/ljharb"

      }

    },

    "node\_modules/http-errors": {

      "version": "2.0.0",

      "resolved": "https://registry.npmjs.org/http-errors/-/http-errors-2.0.0.tgz",

      "integrity": "sha512-FtwrG/euBzaEjYeRqOgly7G0qviiXoJWnvEH2Z1plBdXgbyjv34pHTSb9zoeHMyDy33+DWy5Wt9Wo+TURtOYSQ==",

      "dependencies": {

        "depd": "2.0.0",

        "inherits": "2.0.4",

        "setprototypeof": "1.2.0",

        "statuses": "2.0.1",

        "toidentifier": "1.0.1"

      },

      "engines": {

        "node": ">= 0.8"

      }

    },

    "node\_modules/http-errors/node\_modules/statuses": {

      "version": "2.0.1",

      "resolved": "https://registry.npmjs.org/statuses/-/statuses-2.0.1.tgz",

      "integrity": "sha512-RwNA9Z/7PrK06rYLIzFMlaF+l73iwpzsqRIFgbMLbTcLD6cOao82TaWefPXQvB2fOC4AjuYSEndS7N/mTCbkdQ==",

      "engines": {

        "node": ">= 0.8"

      }

    },

    "node\_modules/iconv-lite": {

      "version": "0.4.24",

      "resolved": "https://registry.npmjs.org/iconv-lite/-/iconv-lite-0.4.24.tgz",

      "integrity": "sha512-v3MXnZAcvnywkTUEZomIActle7RXXeedOR31wwl7VlyoXO4Qi9arvSenNQWne1TcRwhCL1HwLI21bEqdpj8/rA==",

      "dependencies": {

        "safer-buffer": ">= 2.1.2 < 3"

      },

      "engines": {

        "node": ">=0.10.0"

      }

    },

    "node\_modules/inherits": {

      "version": "2.0.4",

      "resolved": "https://registry.npmjs.org/inherits/-/inherits-2.0.4.tgz",

      "integrity": "sha512-k/vGaX4/Yla3WzyMCvTQOXYeIHvqOKtnqBduzTHpzpQZzAskKMhZ2K+EnBiSM9zGSoIFeMpXKxa4dYeZIQqewQ=="

    },

    "node\_modules/ipaddr.js": {

      "version": "1.9.1",

      "resolved": "https://registry.npmjs.org/ipaddr.js/-/ipaddr.js-1.9.1.tgz",

      "integrity": "sha512-0KI/607xoxSToH7GjN1FfSbLoU0+btTicjsQSWQlh/hZykN8KpmMf7uYwPW3R+akZ6R/w18ZlXSHBYXiYUPO3g==",

      "engines": {

        "node": ">= 0.10"

      }

    },

    "node\_modules/media-typer": {

      "version": "0.3.0",

      "resolved": "https://registry.npmjs.org/media-typer/-/media-typer-0.3.0.tgz",

      "integrity": "sha512-dq+qelQ9akHpcOl/gUVRTxVIOkAJ1wR3QAvb4RsVjS8oVoFjDGTc679wJYmUmknUF5HwMLOgb5O+a3KxfWapPQ==",

      "engines": {

        "node": ">= 0.6"

      }

    },

    "node\_modules/merge-descriptors": {

      "version": "1.0.1",

      "resolved": "https://registry.npmjs.org/merge-descriptors/-/merge-descriptors-1.0.1.tgz",

      "integrity": "sha512-cCi6g3/Zr1iqQi6ySbseM1Xvooa98N0w31jzUYrXPX2xqObmFGHJ0tQ5u74H3mVh7wLouTseZyYIq39g8cNp1w=="

    },

    "node\_modules/methods": {

      "version": "1.1.2",

      "resolved": "https://registry.npmjs.org/methods/-/methods-1.1.2.tgz",

      "integrity": "sha512-iclAHeNqNm68zFtnZ0e+1L2yUIdvzNoauKU4WBA3VvH/vPFieF7qfRlwUZU+DA9P9bPXIS90ulxoUoCH23sV2w==",

      "engines": {

        "node": ">= 0.6"

      }

    },

    "node\_modules/mime": {

      "version": "1.6.0",

      "resolved": "https://registry.npmjs.org/mime/-/mime-1.6.0.tgz",

      "integrity": "sha512-x0Vn8spI+wuJ1O6S7gnbaQg8Pxh4NNHb7KSINmEWKiPE4RKOplvijn+NkmYmmRgP68mc70j2EbeTFRsrswaQeg==",

      "bin": {

        "mime": "cli.js"

      },

      "engines": {

        "node": ">=4"

      }

    },

    "node\_modules/mime-db": {

      "version": "1.52.0",

      "resolved": "https://registry.npmjs.org/mime-db/-/mime-db-1.52.0.tgz",

      "integrity": "sha512-sPU4uV7dYlvtWJxwwxHD0PuihVNiE7TyAbQ5SWxDCB9mUYvOgroQOwYQQOKPJ8CIbE+1ETVlOoK1UC2nU3gYvg==",

      "engines": {

        "node": ">= 0.6"

      }

    },

    "node\_modules/mime-types": {

      "version": "2.1.35",

      "resolved": "https://registry.npmjs.org/mime-types/-/mime-types-2.1.35.tgz",

      "integrity": "sha512-ZDY+bPm5zTTF+YpCrAU9nK0UgICYPT0QtT1NZWFv4s++TNkcgVaT0g6+4R2uI4MjQjzysHB1zxuWL50hzaeXiw==",

      "dependencies": {

        "mime-db": "1.52.0"

      },

      "engines": {

        "node": ">= 0.6"

      }

    },

    "node\_modules/ms": {

      "version": "2.0.0",

      "resolved": "https://registry.npmjs.org/ms/-/ms-2.0.0.tgz",

      "integrity": "sha512-Tpp60P6IUJDTuOq/5Z8cdskzJujfwqfOTkrwIwj7IRISpnkJnT6SyJ4PCPnGMoFjC9ddhal5KVIYtAt97ix05A=="

    },

    "node\_modules/negotiator": {

      "version": "0.6.3",

      "resolved": "https://registry.npmjs.org/negotiator/-/negotiator-0.6.3.tgz",

      "integrity": "sha512-+EUsqGPLsM+j/zdChZjsnX51g4XrHFOIXwfnCVPGlQk/k5giakcKsuxCObBRu6DSm9opw/O6slWbJdghQM4bBg==",

      "engines": {

        "node": ">= 0.6"

      }

    },

    "node\_modules/object-inspect": {

      "version": "1.12.3",

      "resolved": "https://registry.npmjs.org/object-inspect/-/object-inspect-1.12.3.tgz",

      "integrity": "sha512-geUvdk7c+eizMNUDkRpW1wJwgfOiOeHbxBR/hLXK1aT6zmVSO0jsQcs7fj6MGw89jC/cjGfLcNOrtMYtGqm81g==",

      "funding": {

        "url": "https://github.com/sponsors/ljharb"

      }

    },

    "node\_modules/on-finished": {

      "version": "2.4.1",

      "resolved": "https://registry.npmjs.org/on-finished/-/on-finished-2.4.1.tgz",

      "integrity": "sha512-oVlzkg3ENAhCk2zdv7IJwd/QUD4z2RxRwpkcGY8psCVcCYZNq4wYnVWALHM+brtuJjePWiYF/ClmuDr8Ch5+kg==",

      "dependencies": {

        "ee-first": "1.1.1"

      },

      "engines": {

        "node": ">= 0.8"

      }

    },

    "node\_modules/options": {

      "version": "0.0.6",

      "resolved": "https://registry.npmjs.org/options/-/options-0.0.6.tgz",

      "integrity": "sha512-bOj3L1ypm++N+n7CEbbe473A414AB7z+amKYshRb//iuL3MpdDCLhPnw6aVTdKB9g5ZRVHIEp8eUln6L2NUStg==",

      "engines": {

        "node": ">=0.4.0"

      }

    },

    "node\_modules/parseurl": {

      "version": "1.3.3",

      "resolved": "https://registry.npmjs.org/parseurl/-/parseurl-1.3.3.tgz",

      "integrity": "sha512-CiyeOxFT/JZyN5m0z9PfXw4SCBJ6Sygz1Dpl0wqjlhDEGGBP1GnsUVEL0p63hoG1fcj3fHynXi9NYO4nWOL+qQ==",

      "engines": {

        "node": ">= 0.8"

      }

    },

    "node\_modules/path-to-regexp": {

      "version": "0.1.7",

      "resolved": "https://registry.npmjs.org/path-to-regexp/-/path-to-regexp-0.1.7.tgz",

      "integrity": "sha512-5DFkuoqlv1uYQKxy8omFBeJPQcdoE07Kv2sferDCrAq1ohOU+MSDswDIbnx3YAM60qIOnYa53wBhXW0EbMonrQ=="

    },

    "node\_modules/proxy-addr": {

      "version": "2.0.7",

      "resolved": "https://registry.npmjs.org/proxy-addr/-/proxy-addr-2.0.7.tgz",

      "integrity": "sha512-llQsMLSUDUPT44jdrU/O37qlnifitDP+ZwrmmZcoSKyLKvtZxpyV0n2/bD/N4tBAAZ/gJEdZU7KMraoK1+XYAg==",

      "dependencies": {

        "forwarded": "0.2.0",

        "ipaddr.js": "1.9.1"

      },

      "engines": {

        "node": ">= 0.10"

      }

    },

    "node\_modules/qs": {

      "version": "6.11.0",

      "resolved": "https://registry.npmjs.org/qs/-/qs-6.11.0.tgz",

      "integrity": "sha512-MvjoMCJwEarSbUYk5O+nmoSzSutSsTwF85zcHPQ9OrlFoZOYIjaqBAJIqIXjptyD5vThxGq52Xu/MaJzRkIk4Q==",

      "dependencies": {

        "side-channel": "^1.0.4"

      },

      "engines": {

        "node": ">=0.6"

      },

      "funding": {

        "url": "https://github.com/sponsors/ljharb"

      }

    },

    "node\_modules/range-parser": {

      "version": "1.2.1",

      "resolved": "https://registry.npmjs.org/range-parser/-/range-parser-1.2.1.tgz",

      "integrity": "sha512-Hrgsx+orqoygnmhFbKaHE6c296J+HTAQXoxEF6gNupROmmGJRoyzfG3ccAveqCBrwr/2yxQ5BVd/GTl5agOwSg==",

      "engines": {

        "node": ">= 0.6"

      }

    },

    "node\_modules/raw-body": {

      "version": "2.5.2",

      "resolved": "https://registry.npmjs.org/raw-body/-/raw-body-2.5.2.tgz",

      "integrity": "sha512-8zGqypfENjCIqGhgXToC8aB2r7YrBX+AQAfIPs/Mlk+BtPTztOvTS01NRW/3Eh60J+a48lt8qsCzirQ6loCVfA==",

      "dependencies": {

        "bytes": "3.1.2",

        "http-errors": "2.0.0",

        "iconv-lite": "0.4.24",

        "unpipe": "1.0.0"

      },

      "engines": {

        "node": ">= 0.8"

      }

    },

    "node\_modules/safe-buffer": {

      "version": "5.2.1",

      "resolved": "https://registry.npmjs.org/safe-buffer/-/safe-buffer-5.2.1.tgz",

      "integrity": "sha512-rp3So07KcdmmKbGvgaNxQSJr7bGVSVk5S9Eq1F+ppbRo70+YeaDxkw5Dd8NPN+GD6bjnYm2VuPuCXmpuYvmCXQ==",

      "funding": [

        {

          "type": "github",

          "url": "https://github.com/sponsors/feross"

        },

        {

          "type": "patreon",

          "url": "https://www.patreon.com/feross"

        },

        {

          "type": "consulting",

          "url": "https://feross.org/support"

        }

      ]

    },

    "node\_modules/safer-buffer": {

      "version": "2.1.2",

      "resolved": "https://registry.npmjs.org/safer-buffer/-/safer-buffer-2.1.2.tgz",

      "integrity": "sha512-YZo3K82SD7Riyi0E1EQPojLz7kpepnSQI9IyPbHHg1XXXevb5dJI7tpyN2ADxGcQbHG7vcyRHk0cbwqcQriUtg=="

    },

    "node\_modules/send": {

      "version": "0.17.2",

      "resolved": "https://registry.npmjs.org/send/-/send-0.17.2.tgz",

      "integrity": "sha512-UJYB6wFSJE3G00nEivR5rgWp8c2xXvJ3OPWPhmuteU0IKj8nKbG3DrjiOmLwpnHGYWAVwA69zmTm++YG0Hmwww==",

      "dependencies": {

        "debug": "2.6.9",

        "depd": "~1.1.2",

        "destroy": "~1.0.4",

        "encodeurl": "~1.0.2",

        "escape-html": "~1.0.3",

        "etag": "~1.8.1",

        "fresh": "0.5.2",

        "http-errors": "1.8.1",

        "mime": "1.6.0",

        "ms": "2.1.3",

        "on-finished": "~2.3.0",

        "range-parser": "~1.2.1",

        "statuses": "~1.5.0"

      },

      "engines": {

        "node": ">= 0.8.0"

      }

    },

    "node\_modules/send/node\_modules/depd": {

      "version": "1.1.2",

      "resolved": "https://registry.npmjs.org/depd/-/depd-1.1.2.tgz",

      "integrity": "sha512-7emPTl6Dpo6JRXOXjLRxck+FlLRX5847cLKEn00PLAgc3g2hTZZgr+e4c2v6QpSmLeFP3n5yUo7ft6avBK/5jQ==",

      "engines": {

        "node": ">= 0.6"

      }

    },

    "node\_modules/send/node\_modules/destroy": {

      "version": "1.0.4",

      "resolved": "https://registry.npmjs.org/destroy/-/destroy-1.0.4.tgz",

      "integrity": "sha512-3NdhDuEXnfun/z7x9GOElY49LoqVHoGScmOKwmxhsS8N5Y+Z8KyPPDnaSzqWgYt/ji4mqwfTS34Htrk0zPIXVg=="

    },

    "node\_modules/send/node\_modules/http-errors": {

      "version": "1.8.1",

      "resolved": "https://registry.npmjs.org/http-errors/-/http-errors-1.8.1.tgz",

      "integrity": "sha512-Kpk9Sm7NmI+RHhnj6OIWDI1d6fIoFAtFt9RLaTMRlg/8w49juAStsrBgp0Dp4OdxdVbRIeKhtCUvoi/RuAhO4g==",

      "dependencies": {

        "depd": "~1.1.2",

        "inherits": "2.0.4",

        "setprototypeof": "1.2.0",

        "statuses": ">= 1.5.0 < 2",

        "toidentifier": "1.0.1"

      },

      "engines": {

        "node": ">= 0.6"

      }

    },

    "node\_modules/send/node\_modules/ms": {

      "version": "2.1.3",

      "resolved": "https://registry.npmjs.org/ms/-/ms-2.1.3.tgz",

      "integrity": "sha512-6FlzubTLZG3J2a/NVCAleEhjzq5oxgHyaCU9yYXvcLsvoVaHJq/s5xXI6/XXP6tz7R9xAOtHnSO/tXtF3WRTlA=="

    },

    "node\_modules/send/node\_modules/on-finished": {

      "version": "2.3.0",

      "resolved": "https://registry.npmjs.org/on-finished/-/on-finished-2.3.0.tgz",

      "integrity": "sha512-ikqdkGAAyf/X/gPhXGvfgAytDZtDbr+bkNUJ0N9h5MI/dmdgCs3l6hoHrcUv41sRKew3jIwrp4qQDXiK99Utww==",

      "dependencies": {

        "ee-first": "1.1.1"

      },

      "engines": {

        "node": ">= 0.8"

      }

    },

    "node\_modules/serve-static": {

      "version": "1.14.2",

      "resolved": "https://registry.npmjs.org/serve-static/-/serve-static-1.14.2.tgz",

      "integrity": "sha512-+TMNA9AFxUEGuC0z2mevogSnn9MXKb4fa7ngeRMJaaGv8vTwnIEkKi+QGvPt33HSnf8pRS+WGM0EbMtCJLKMBQ==",

      "dependencies": {

        "encodeurl": "~1.0.2",

        "escape-html": "~1.0.3",

        "parseurl": "~1.3.3",

        "send": "0.17.2"

      },

      "engines": {

        "node": ">= 0.8.0"

      }

    },

    "node\_modules/setprototypeof": {

      "version": "1.2.0",

      "resolved": "https://registry.npmjs.org/setprototypeof/-/setprototypeof-1.2.0.tgz",

      "integrity": "sha512-E5LDX7Wrp85Kil5bhZv46j8jOeboKq5JMmYM3gVGdGH8xFpPWXUMsNrlODCrkoxMEeNi/XZIwuRvY4XNwYMJpw=="

    },

    "node\_modules/side-channel": {

      "version": "1.0.4",

      "resolved": "https://registry.npmjs.org/side-channel/-/side-channel-1.0.4.tgz",

      "integrity": "sha512-q5XPytqFEIKHkGdiMIrY10mvLRvnQh42/+GoBlFW3b2LXLE2xxJpZFdm94we0BaoV3RwJyGqg5wS7epxTv0Zvw==",

      "dependencies": {

        "call-bind": "^1.0.0",

        "get-intrinsic": "^1.0.2",

        "object-inspect": "^1.9.0"

      },

      "funding": {

        "url": "https://github.com/sponsors/ljharb"

      }

    },

    "node\_modules/statuses": {

      "version": "1.5.0",

      "resolved": "https://registry.npmjs.org/statuses/-/statuses-1.5.0.tgz",

      "integrity": "sha512-OpZ3zP+jT1PI7I8nemJX4AKmAX070ZkYPVWV/AaKTJl+tXCTGyVdC1a4SL8RUQYEwk/f34ZX8UTykN68FwrqAA==",

      "engines": {

        "node": ">= 0.6"

      }

    },

    "node\_modules/toidentifier": {

      "version": "1.0.1",

      "resolved": "https://registry.npmjs.org/toidentifier/-/toidentifier-1.0.1.tgz",

      "integrity": "sha512-o5sSPKEkg/DIQNmH43V0/uerLrpzVedkUh8tGNvaeXpfpuwjKenlSox/2O/BTlZUtEe+JG7s5YhEz608PlAHRA==",

      "engines": {

        "node": ">=0.6"

      }

    },

    "node\_modules/type-is": {

      "version": "1.6.18",

      "resolved": "https://registry.npmjs.org/type-is/-/type-is-1.6.18.tgz",

      "integrity": "sha512-TkRKr9sUTxEH8MdfuCSP7VizJyzRNMjj2J2do2Jr3Kym598JVdEksuzPQCnlFPW4ky9Q+iA+ma9BGm06XQBy8g==",

      "dependencies": {

        "media-typer": "0.3.0",

        "mime-types": "~2.1.24"

      },

      "engines": {

        "node": ">= 0.6"

      }

    },

    "node\_modules/ultron": {

      "version": "1.0.2",

      "resolved": "https://registry.npmjs.org/ultron/-/ultron-1.0.2.tgz",

      "integrity": "sha512-QMpnpVtYaWEeY+MwKDN/UdKlE/LsFZXM5lO1u7GaZzNgmIbGixHEmVMIKT+vqYOALu3m5GYQy9kz4Xu4IVn7Ow=="

    },

    "node\_modules/unpipe": {

      "version": "1.0.0",

      "resolved": "https://registry.npmjs.org/unpipe/-/unpipe-1.0.0.tgz",

      "integrity": "sha512-pjy2bYhSsufwWlKwPc+l3cN7+wuJlK6uz0YdJEOlQDbl6jo/YlPi4mb8agUkVC8BF7V8NuzeyPNqRksA3hztKQ==",

      "engines": {

        "node": ">= 0.8"

      }

    },

    "node\_modules/utils-merge": {

      "version": "1.0.1",

      "resolved": "https://registry.npmjs.org/utils-merge/-/utils-merge-1.0.1.tgz",

      "integrity": "sha512-pMZTvIkT1d+TFGvDOqodOclx0QWkkgi6Tdoa8gC8ffGAAqz9pzPTZWAybbsHHoED/ztMtkv/VoYTYyShUn81hA==",

      "engines": {

        "node": ">= 0.4.0"

      }

    },

    "node\_modules/vary": {

      "version": "1.1.2",

      "resolved": "https://registry.npmjs.org/vary/-/vary-1.1.2.tgz",

      "integrity": "sha512-BNGbWLfd0eUPabhkXUVm0j8uuvREyTh5ovRa/dyow/BqAbZJyC+5fU+IzQOzmAKzYqYRAISoRhdQr3eIZ/PXqg==",

      "engines": {

        "node": ">= 0.8"

      }

    },

    "node\_modules/ws": {

      "version": "1.1.5",

      "resolved": "https://registry.npmjs.org/ws/-/ws-1.1.5.tgz",

      "integrity": "sha512-o3KqipXNUdS7wpQzBHSe180lBGO60SoK0yVo3CYJgb2MkobuWuBX6dhkYP5ORCLd55y+SaflMOV5fqAB53ux4w==",

      "dependencies": {

        "options": ">=0.0.5",

        "ultron": "1.0.x"

      }

    }

  }

}

# Изменения кода

var calculateHash = (index, previousHash, timestamp, data, nonce) = > {

    var hash = CryptoJS.SHA256(index + previousHash + timestamp + data + nonce).toString();

    var decimalSum = 0;

    for (var i = 0; i < hash.length; i++) {

        decimalSum += parseInt(hash[i], 16);

    }

    return decimalSum.toString();

};

В данном коде происходит вычисление хэша (hash) на основе заданных параметров: index (индекс), previousHash (хэш предыдущего блока), timestamp (временная метка), data (данные блока) и nonce (случайное число). Хэш вычисляется с использованием алгоритма SHA-256 (Secure Hash Algorithm 256-bit).

Первая строка кода **var hash = CryptoJS.SHA256(index + previousHash + timestamp + data + nonce).toString();** создает строку, содержащую объединение всех параметров блока, а затем вычисляет хэш этой строки с помощью функции SHA-256 из библиотеки CryptoJS. Результат преобразуется в строку.

Затем следует цикл **for**, который перебирает все символы в полученном хэше. Внутри цикла каждый символ хэша преобразуется в десятичное число с помощью **parseInt(hash[i], 16)**, где **hash[i]** - текущий символ хэша, а **16** указывает, что символы интерпретируются как шестнадцатеричные числа. Затем полученное десятичное число суммируется с предыдущими значениями в переменной **decimalSum**.

В конце функции значение **decimalSum** преобразуется в строку и возвращается как результат вычисления хэша в десятичной системе.

var difficulty = 0;

Теперь difficulty равно нулю, так как нам сейчас не важно количество нулей в начале получаемого хэша, так как идет перевод в другую систему счисления.

var mineBlock = (blockData) => {

    var previousBlock = getLatestBlock();

    var nextIndex = previousBlock.index + 1;

    var nonce = 0;

    var nextTimestamp = new Date().getTime() / 1000;

    var nextHash = calculateHash(nextIndex, previousBlock.hash, nextTimestamp, blockData, nonce);

*while* (nextHash % 37 !== 0) {

        nonce++;

        nextTimestamp = new Date().getTime() / 1000;

        nextHash = calculateHash(nextIndex, previousBlock.hash, nextTimestamp, blockData, nonce);

        console.log("\"index\": " + nextIndex + ", \"previousHash\": " + previousBlock.hash + ", \"timestamp\": " + nextTimestamp + ", \"data\": " + blockData + ", \"hash\": " + nextHash + ", \"difficulty\": " + difficulty + ", \"nonce\": " + nonce);

    }

*return* new Block(nextIndex, previousBlock.hash, nextTimestamp, blockData, nextHash, difficulty, nonce);

};

Цикл будет выполняться до тех пор, пока не будет найдено подходящее значение nonce, удовлетворяющее условию nextHash % 37 === 0. Когда это условие выполнено, цикл завершается, и функция mineBlock возвращает новый блок с найденным значением nonce.

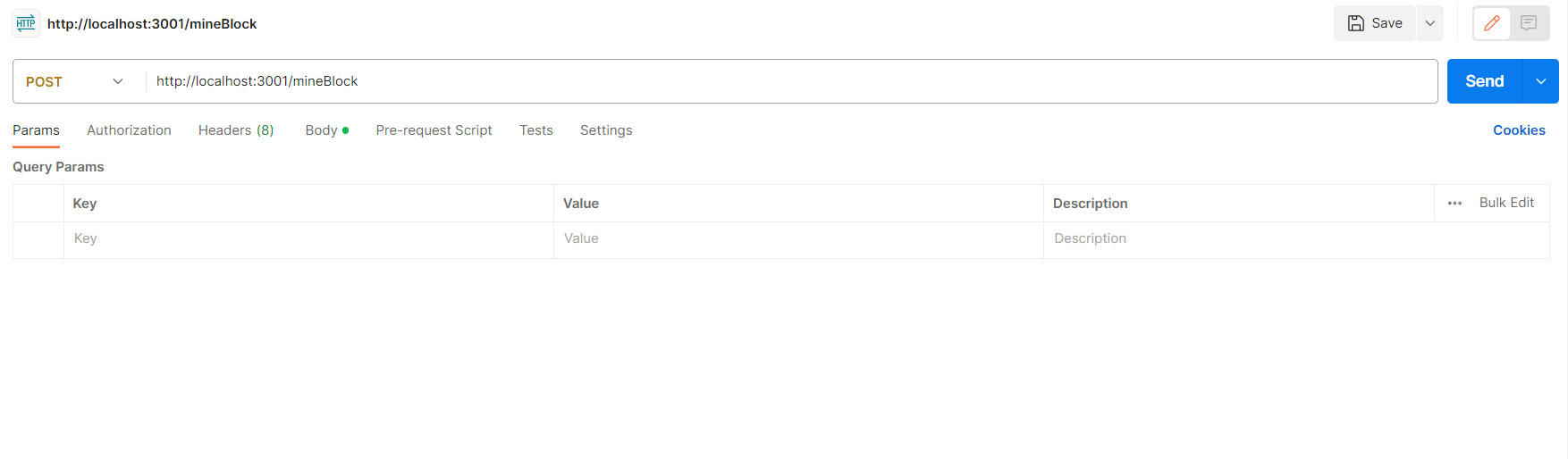
# Результаты

Сервер запускаю через два терминала, со следующими командами:

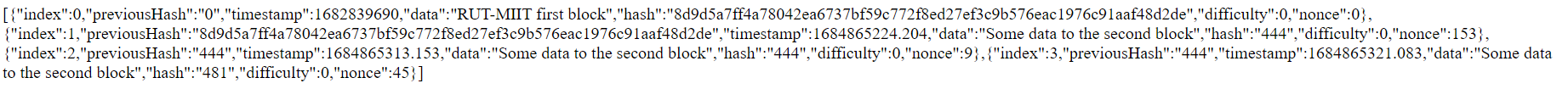
Для первого  
$env:HTTP\_PORT=3001; $env:P2P\_PORT=6001; npm start

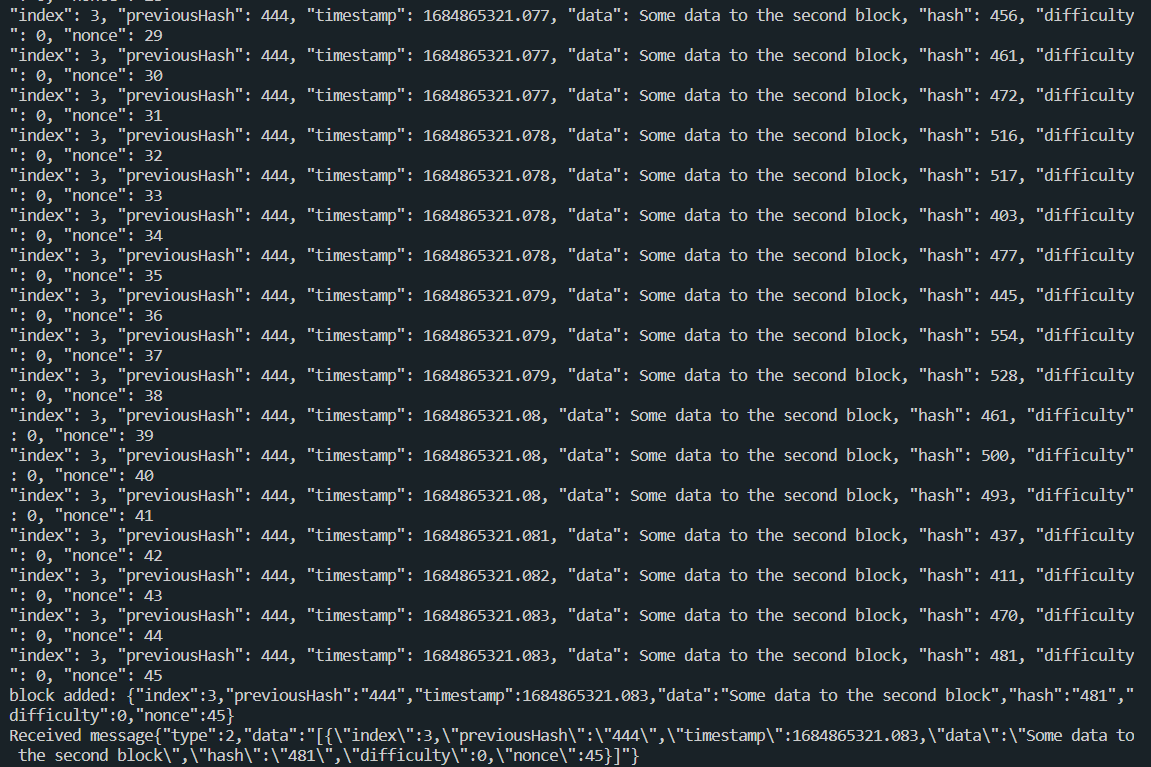
Для второго

$env:HTTP\_PORT=3002; $env:P2P\_PORT=6002; $env:PEERS="ws://192.168.1.62:6001"; npm start

Отправляем запрос в Postman   


На сервере получаем искомый нами хэш, а то есть число делящееся на 37, т.е. подходящее по нашему условию.

  
  
В терминале VS Code получаем следующее:



# Ссылка на GitHub

Ссылка на репотизотрий проекта: <https://github.com/Egorchik44/Blockchain>