

Test Report

Started: 2022-06-04 21:06:34

Suites (6) <div><div></div><div>6 passed</div><div>0 failed</div><div>0 pending</div></div>	Tests (65) <div><div></div><div>65 passed</div><div>0 failed</div><div>0 pending</div></div>
--	---

C:\Users\sarap\Github\Blockchain\cryptochain\util\crypto-hash.test.js		2.758s
cryptoHash()	generates a SHA-256 hashed output	passed0.009s
cryptoHash()	produces the same has with the same input arguments in any order	passed0s
cryptoHash()	produces a unique hash when the properties have changed on an input	passed0.001s

C:\Users\sarap\Github\Blockchain\cryptochain\blockchain\block.test.js		3.362s
Block	has a timestamp, lastHash, hash, nonce, difficulty and data property	passed0.008s
Block > genesis()	returns a Block instance	passed0.001s
Block > genesis()	returns the genesis data	passed0.001s
Block > mineBlock()	returns a Block instance	passed0.001s
Block > mineBlock()	sets the 'lastHash' to be the hash of the 'lastBlock'	passed0.002s
Block > mineBlock()	sets the 'data'	passed0.003s
Block > mineBlock()	sets a 'timestamp'	passed0.001s
Block > mineBlock()	creates a SHA-256 'hash' based on the proper inputs	passed0.001s
Block > mineBlock()	sets a 'hash' that matches the difficulty criteria	passed0.001s
Block > mineBlock()	adjust the difficulty	passed0.001s
Block > adjustDifficulty()	raises the difficulty for a quickly mined block	passed0.001s
Block > adjustDifficulty()	lowers the difficulty for a slowly mined block	passed0s
Block > adjustDifficulty()	has a lower limit of 1	passed0s

C:\Users\sarap\Github\Blockchain\cryptochain\blockchain\index.test.js		3.452s
Blockchain	contains a 'chain' Array instance	passed0.005s
Blockchain	starts with the genesis block	passed0.003s
Blockchain	adds a new block to the chain	passed0.002s
Blockchain > isValidChain() > when the chain does not start with the genesis block	returns false	passed0.002s
Blockchain > isValidChain() > when the chain starts with the genesis block and has multiple blocks > and a lastHash reference has changed	returns false	passed0.002s
Blockchain > isValidChain() > when the chain starts with the genesis block and has multiple blocks > and the chain contains a block with an invalid field	returns false	passed0.002s
Blockchain > isValidChain() > when the chain starts with the genesis block and has multiple blocks > and the chain contains a block with a jumped difficulty	returns false	passed0.001s
Blockchain > isValidChain() > when the chain starts with the genesis block and has multiple blocks > and the chain does not contain any invalid blocks	returns true	passed0.002s
Blockchain > replaceChain() > when the new chain is not longer	does not replace the chain	passed0.002s
Blockchain > replaceChain() > when the new chain is not longer	logs an error	passed0.002s
Blockchain > replaceChain() > when the new chain is longer > when the new chain is invalid	does not replace the chain	passed0.003s
Blockchain > replaceChain() > when the new chain is longer > when the new chain is invalid	logs an error	passed0.001s
Blockchain > replaceChain() > when the new chain is longer > when the new chain is valid	replaces the chain	passed0.003s
Blockchain > replaceChain() > when the new chain is longer > when the new chain is valid	logs about the new chain replacement	passed0s

C:\Users\sarap\Github\Blockchain\cryptochain\wallet\index.test.js		6.783s
Wallet	has a 'balance'	passed0.177s
Wallet	has a 'publicKey'	passed0.136s
Wallet > signing data	verifies a signature	passed0.78s
Wallet > signing data	does not verify an invalid signature	passed1.15s
Wallet > createTransaction() > and the amount exceeds the balance	throws an error	passed0.162s
Wallet > createTransaction() > and the amount is valid	creates an instance of 'Transaction'	passed0.322s
Wallet > createTransaction() > and the amount is valid	matches the transaction input with the wallet	passed0.369s
Wallet > createTransaction() > and the amount is valid	outputs the amount the recipient	passed0.314s

C:\Users\sarap\Github\Blockchain\cryptochain\wallet\transaction.test.js		11.672s
Transaction	has an 'id'	passed0.362s
Transaction > outputMap	has an 'outputMap'	passed0.269s
Transaction > outputMap	outputs the amount to the recipient	passed0.301s
Transaction > outputMap	outputs the remaining balance for the 'senderWallet'	passed0.317s
Transaction > input	has an 'input'	passed0.406s
Transaction > input	has a 'timestamp' in the input	passed0.425s
Transaction > input	sets the 'amount' to the 'senderWallet' balance	passed0.374s
Transaction > input	sets the 'address' to the 'senderWallet' publicKey	passed0.33s
Transaction > input	signs the input	passed0.828s
Transaction > validTransaction() > when the transaction is valid	returns true	passed0.621s
Transaction > validTransaction() > when the transaction is invalid > and a transaction outputMap value is invalid	returns false and logs an error	passed0.22s
Transaction > validTransaction() > when the transaction is invalid > and the transaction input signature is invalid	returns false and logs an error	passed0.794s
Transaction > update() > and the amount is invalid	throws an error	passed0.224s
Transaction > update() > and the amount is valid	outputs the amount to the next recipient	passed0.325s
Transaction > update() > and the amount is valid	subtracts the amount from the original sender output amount	passed0.33s
Transaction > update() > and the amount is valid	maintains a total output that matches the input amount	passed0.34s
Transaction > update() > and the amount is valid	re-signs the transaction	passed0.356s
Transaction > update() > and the amount is valid > and another update for the same recipient	adds to the recipient amount	passed0.456s
Transaction > update() > and the amount is valid > and another update for the same recipient	subtracts the amount from the original sender output amount	passed0.418s
Transaction > rewardTransaction()	creates a transaction with the reward input	passed0.254s
Transaction > rewardTransaction()	creates one transaction for the miner with the 'MINING_REWARD'	passed0.278s

C:\Users\sarap\Github\Blockchain\cryptochain\wallet\transaction-pool.test.js		13.647s
TransactionPool > setTransaction()	adds a transaction	passed0.319s
TransactionPool > existingTransaction()	returns an existing transaction given an input address	passed0.262s
TransactionPool > validTransactions()	returns valid transaction	passed5.041s
TransactionPool > validTransactions()	logs errors for the invalid transactions	passed3.339s
TransactionPool > clear()	clears the transactions	passed0.147s
TransactionPool > clearBlockchainTransactions()	clears the pool of any existing blockchain transactions	passed1.067s