**Testing**

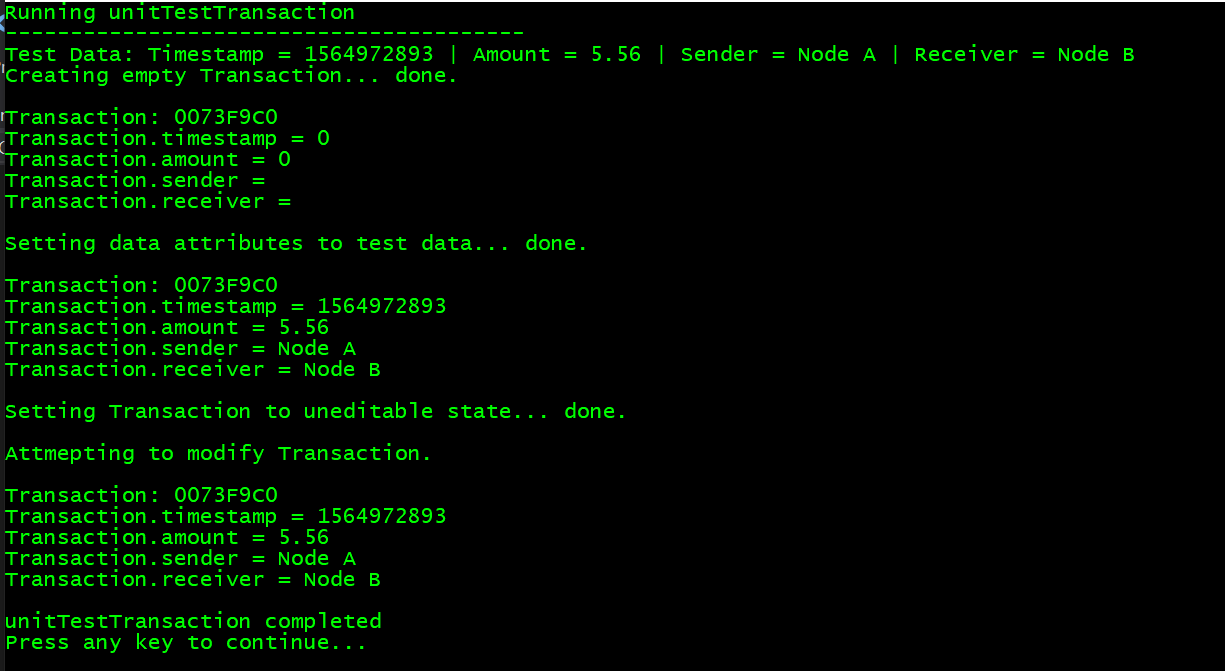
By Eric Brun

**Unit Tests**

**Transaction**

* confirm that all Transaction attributes are set to the correct passed values in the test
* confirm that Transaction attributes cannot be modified after executing makeConstant() method

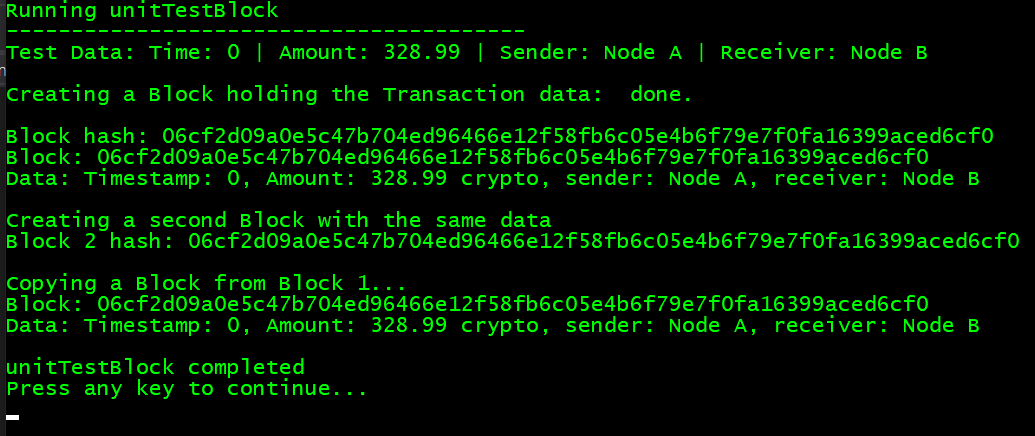
Results:



**Block**

* confirm that the constructor creates a Block instance with correct attribute values
* confirm that CalculateHash() produces a hash string with the correct Nonce
* confirm that the GetData() method returns a Transaction instance with the correct values
* confirm that copying a Block copies all attribute values (hash can vary for improved encryption)

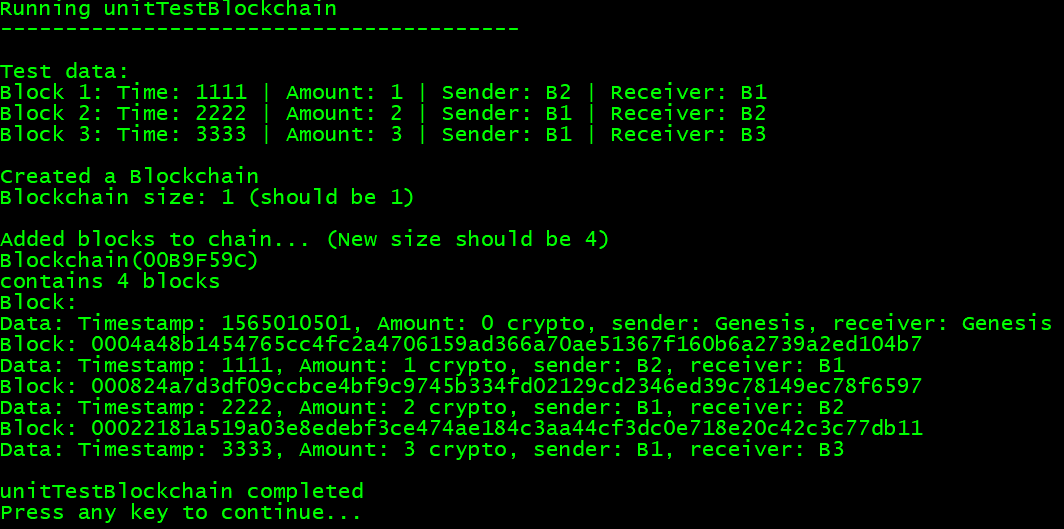
Results:



**Blockchain**

* confirm that the constructor creates a Blockchain with only an origin block
* confirm that adding a Block to the chain creates a new last Block in the chain with the desired Transaction attributes, and the size of the chain increases

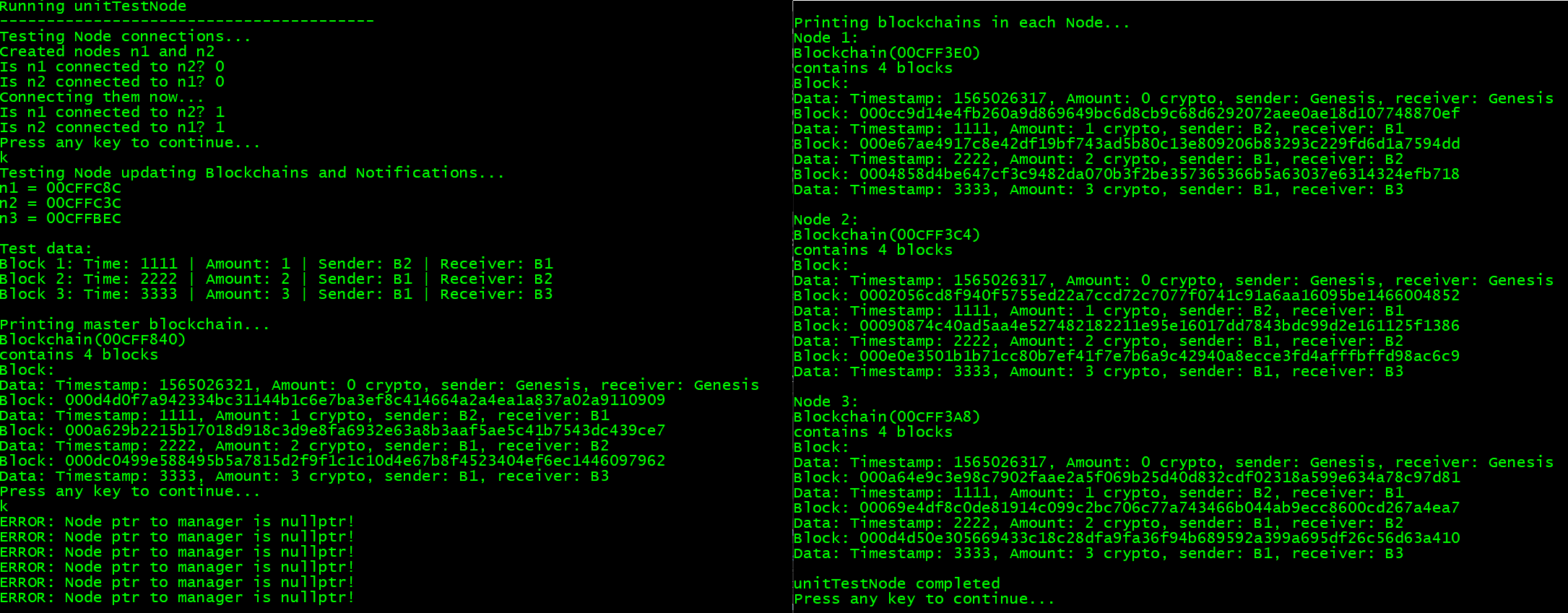
Results:



**Node**

* confirm that getters and setters work
* confirm that when calling node1.addConnection(node2) then both nodes are connected to one another
* confirm Nodes correctly modify their Blockchains and Blockchains have correct Transaction data
* confirm that a Node does not crash the program when a DataManager is not connected
  + instead an appropriate error message is printed

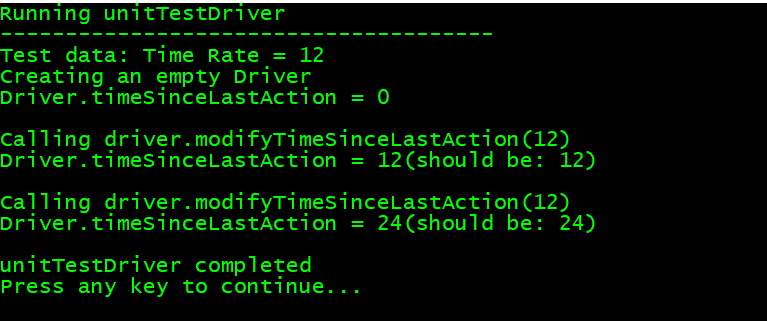
Results:



**Driver**

* confirm that getters and setters work

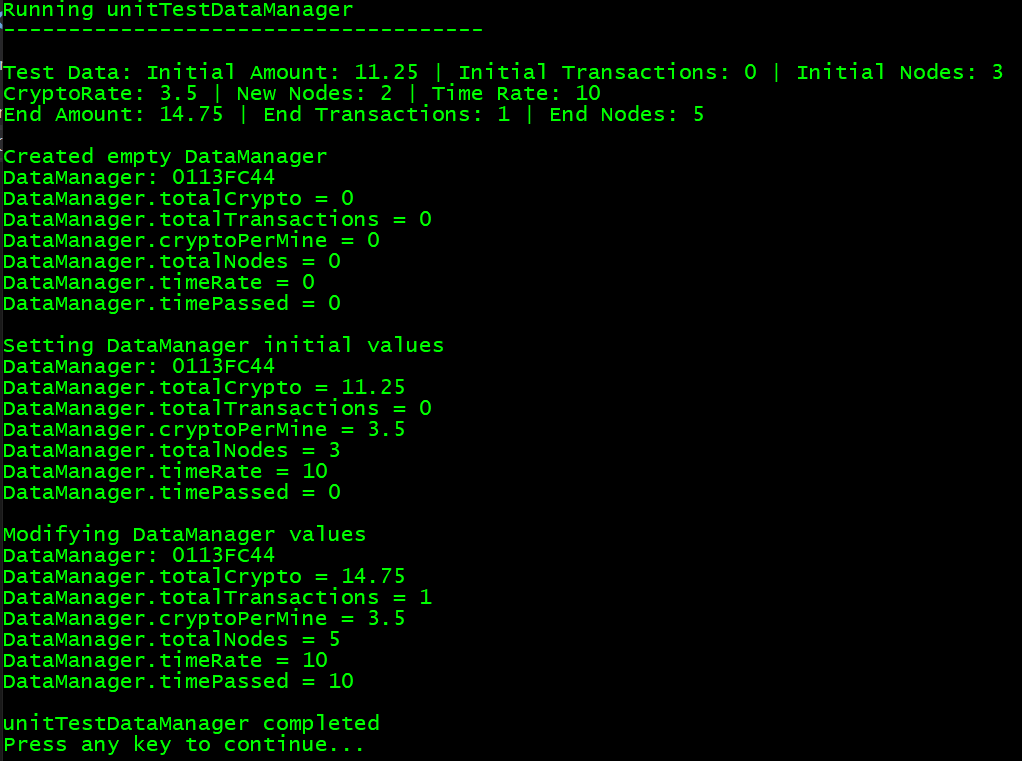
Results:



**DataManager**

-confirm that getters and setters work

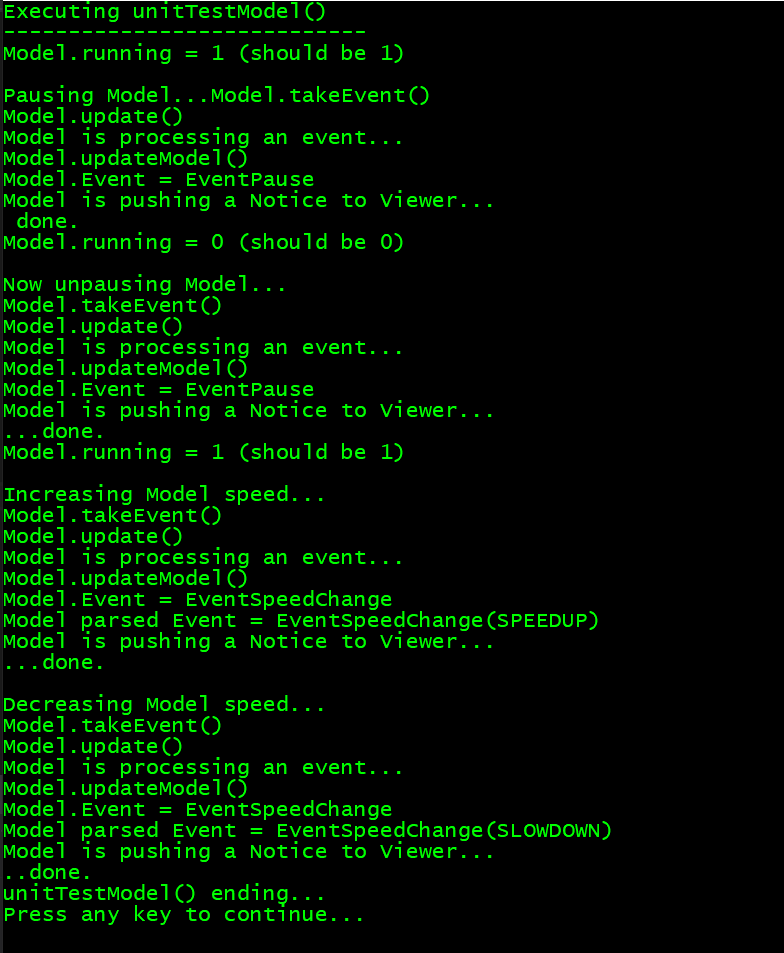
Results:



**Model**

* confirm that when a given Event type is passed to the Model, that it takes the corresponding action

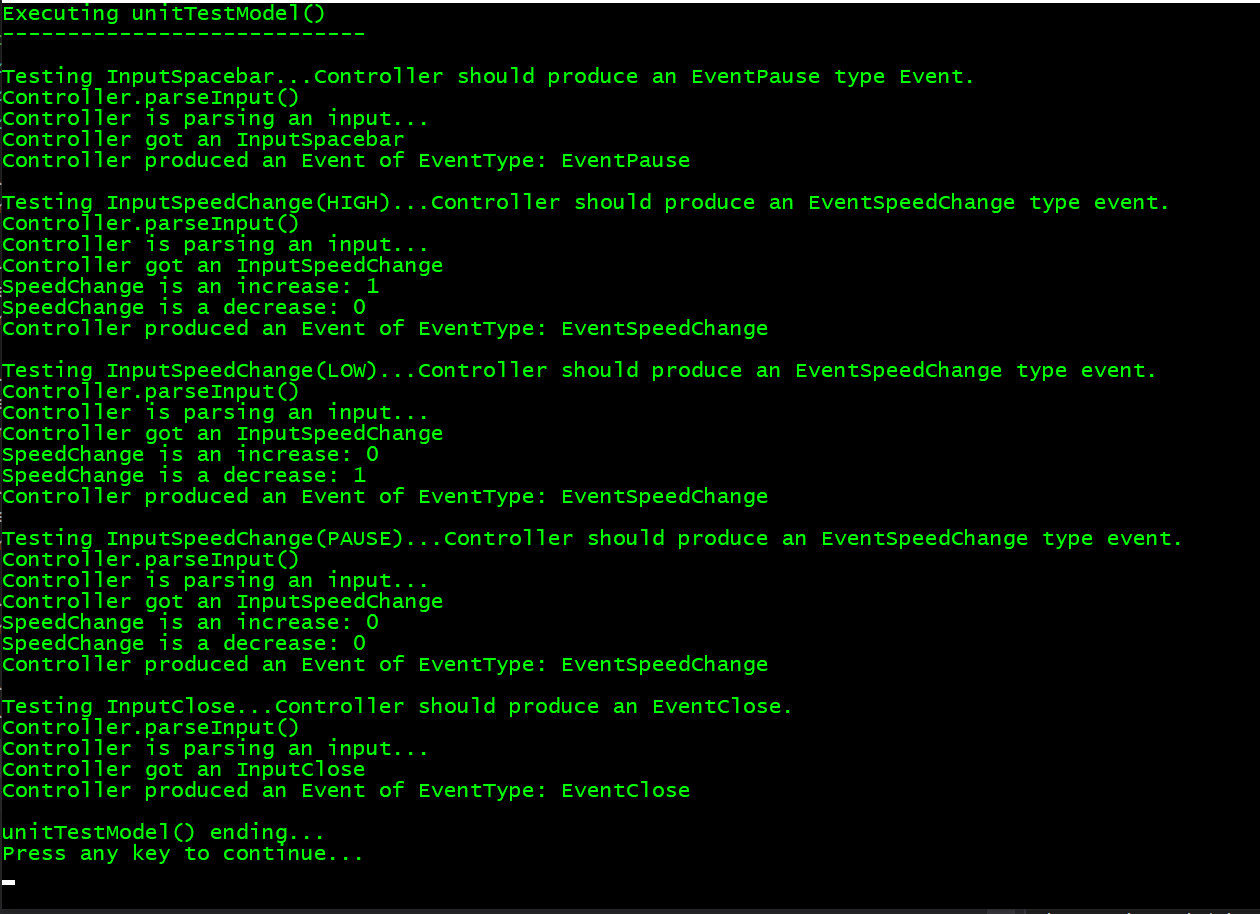
Results:



**Controller**

* confirm that when a given Input type is received by the Controller, then it creates the corresponding Event type

Results:



**System Testing**

**Model – Viewer – Controller system**

* To pass testing, all acceptance cases must be true:

1. When the user clicks the square pause button, the Simulation either pauses and the button turns red or it unpauses and the button turns green. If the simulation is paused, no changes to the Model are made.

True

2. When the user hits the spacebar, the Simulation either pauses and the button turns red or it unpauses and the button turns green. If the simulation is paused, no changes to the Model are made.

True

3. When the user clicks the left arrow or right arrow buttons to the sides of the pause button, the Simulation modifies its update rate to be either faster (displayed rate decreases in value if right arrow clicked) or slower (displayed rate increase sin value if left arrow clicked)

True

4. When enough time has passed since the last Simulation update and the Sim is not paused, then the Simulation Driver takes an action and the Viewer screen updates accordingly.

True

5. When the user presses the escape key, the program shuts down.

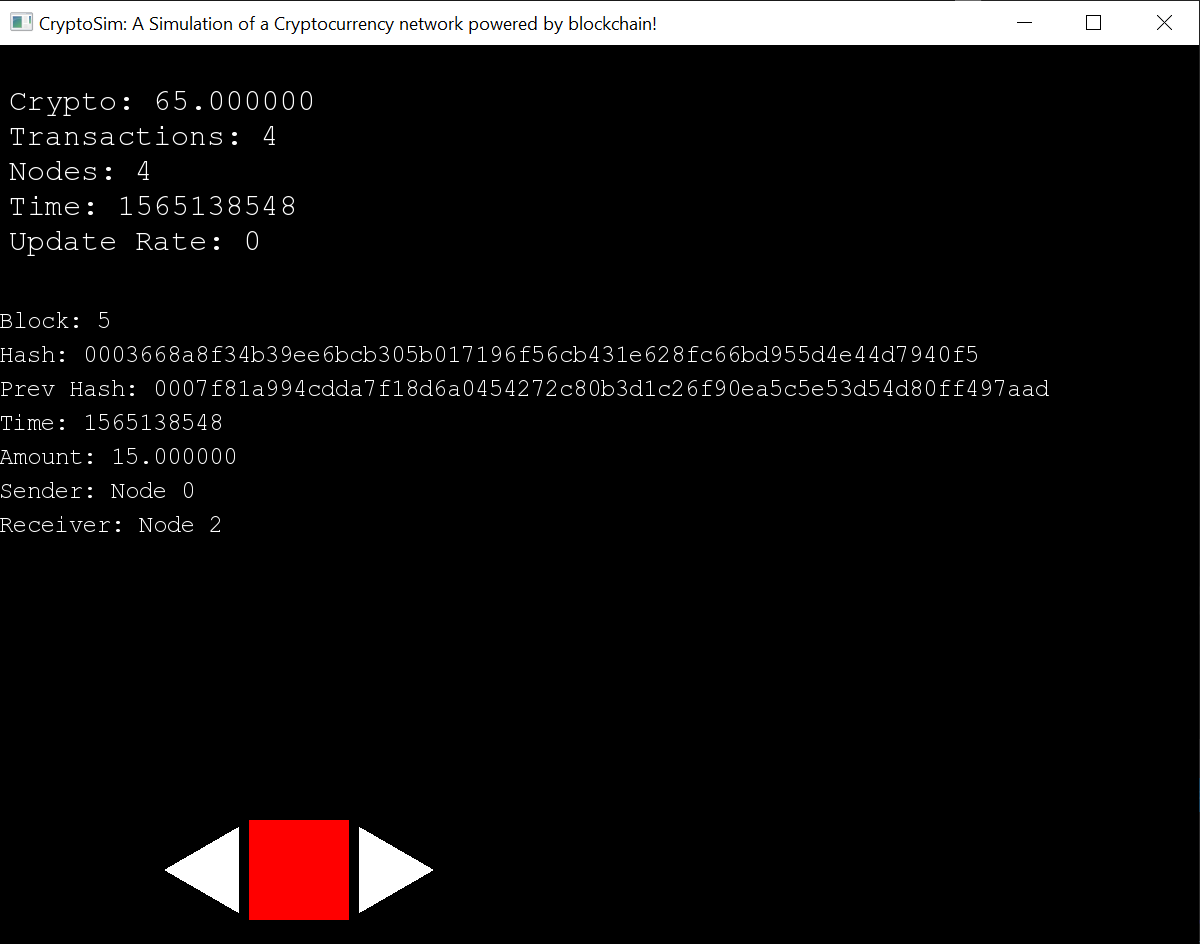
True

6. When the user clicks the X in the top right corner of the display window, the program shuts down.

True

Result screenshots on the following pages.

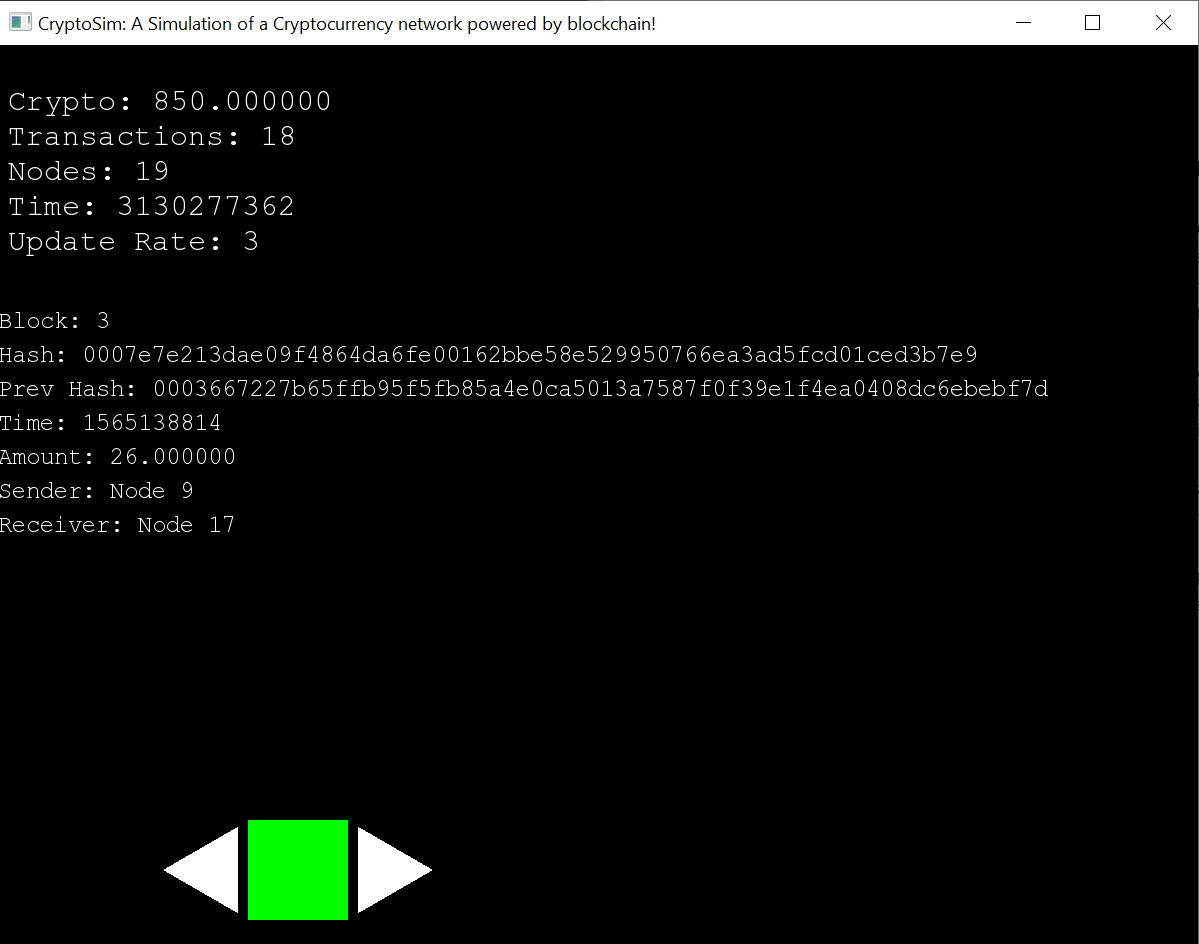
1. Paused state:



2. Running state:



3. Sim with Increased Speed:



4. Sim with Decreased Speed:

