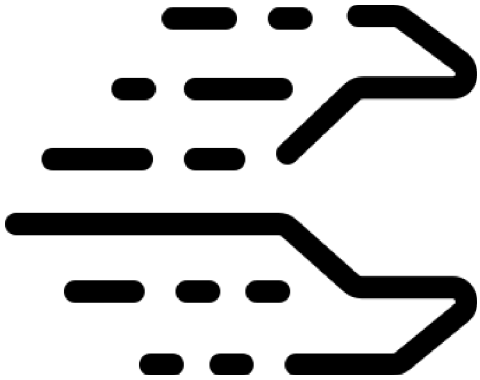
CERTIK

Formal Verification Platform for
Smart Contracts and Blockchain Ecosystems

CertiK Verification Report for Celer

Verification Request date: 2018-08-20
Company Website: https://www.celer.network/



Summary

This is the report for smart contract verification service on CelerTimelock.sol, CelerToken.sol and CelerCrowdsale.sol from Celer Network. The goal of the audition is to guarantee that verified smart contracts are robust enough to avoid potentially unexpected loopholes.

The hash of source codes is appended to the appendix of this report.

Conclusion

PASS

Our formal verification engine concludes that Celer smart contracts meet most of the specification, with 100% code coverage. CertiK believes these contracts are trustworthy and hack-resistant.

Details

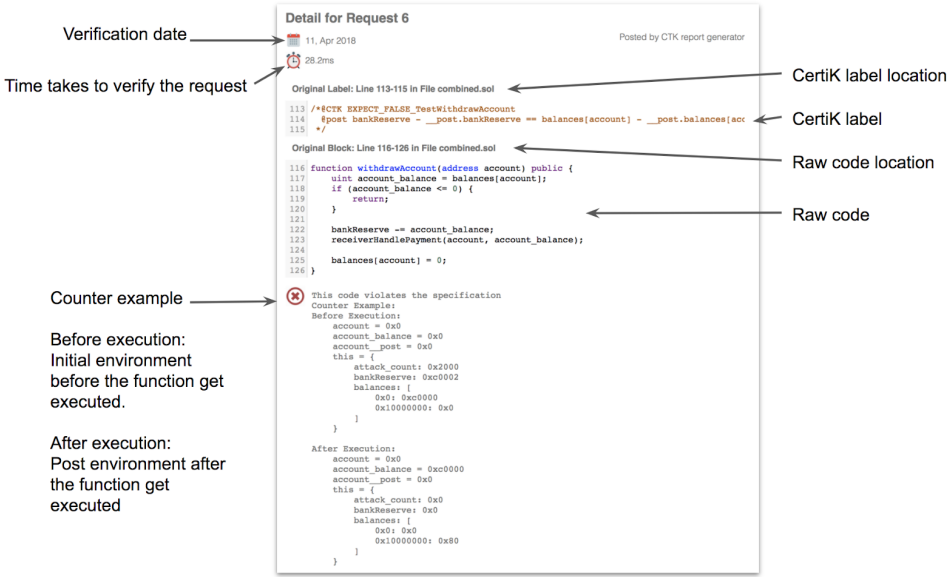
1. Vulnerability

CertiK applied 100% covered smart labels on the source code to detect 2 types of errors: function correctness, and integer overflow. For each failed verification request, CertiK categorized it into 3 buckets: Critical, Medium and Low, based on its severity. If there is any issue falling into the Critical and/or Medium bucket, CertiK will push back and require the client to update the source code to meet criteria.

Severity	Issues Not Found
Critical	Not found
Medium	Not found
Low	Not found

Property checking and function correctness verification were applied to the source code. Our formal verification engine concludes that the Celer smart contracts implementation meets 100% of the specification. There is no production security concerns.

2. How to Read



3. Disclaimer

This report is subject to the terms and conditions (including without limitation, description of the services, confidentiality, disclaimer and limitation of liability) set forth in the Verification Services Agreement between CertiK and Celer Network, or the scope of verification, and terms and conditions provided to Celer in connection with this verification. No third party shall be entitled to rely on this report or have any legal or equitable right, benefit or remedy of any nature whatsoever, under or by reason of this report. CertiK assumes no liability to any third party because of reliance on this report.

Appendix: Source Code

CelerTimelock.sol[1]

```
$ shasum -a 256 CelerTimelock.sol
35ef01d7ae56c392c42106d5370c7f27d2e8bc42fe3b400f7acef1e62a3c33b3  CelerTimelock.sol
```

CelerToken.sol

```
$ shasum -a 256 CelerToken.sol
3a4485b6af2c6f7e426ebd2df7d743f6c43096cda96fcac6d0b503d40fc12f8a  CelerToken.sol
```

CelerCrowdsale.sol

```
$ shasum -a 256 CelerCrowdsale.sol
6da7b5ea73a115430d3456aa377cd795feb4d5543fa89076a56842faf3621de9  CelerCrowdsale.sol
```

[1] this includes all the imported library codes.

CertiK

Certi Request Report

65 out of 65 specs are satisfied.

Detail for Request 0: SafeMath_mul

08, Sep 2018

353.8ms

Posted by CTK report generator

Line 14-23 in File CelerToken.sol

```
NaN | /*@CTK SafeMath_mul
NaN |   @tag spec
NaN |   @post __reverted == __has_assertion_failure
NaN |   @post __has_assertion_failure == __has_overflow
NaN |   @post __reverted == false -> c == a * b
NaN |   @post a == 0 -> c == 0
NaN |   @post msg == msg__post
NaN |   @post (a > 0 && (a * b / a != b)) == __has_assertion_failure
```

```
NaN   @post __addr_map == __addr_map__post
NaN */
```

Line 24-35 in File CelerToken.sol

```
NaN function mul(uint256 a, uint256 b) internal pure returns (uint256 c) {
NaN   // Gas optimization: this is cheaper than asserting 'a' not being zero, but the
NaN   // benefit is lost if 'b' is also tested.
NaN   // See: https://github.com/OpenZeppelin/openzeppelin-solidity/pull/522
NaN   if (a == 0) {
NaN     return 0;
NaN   }
NaN
NaN   c = a * b;
NaN   assert(c / a == b);
NaN   return c;
NaN }
```

The code meets the specification

Detail for Request 1: SafeMath_div

08, Sep 2018

Posted by CTK report generator

0.4ms

Line 40-48 in File CelerToken.sol

```
NaN /*@CTK SafeMath_div
NaN   @tag spec
NaN   @post __reverted == __has_assertion_failure
NaN   @post b == 0 -> __reverted == true  // solidity throws on 0.
NaN   @post __has_overflow == true -> __has_assertion_failure == true
NaN   @post __reverted == false -> __return == a / b
NaN   @post msg == msg__post
NaN   @post __addr_map == __addr_map__post
NaN */
```

Line 49-54 in File CelerToken.sol

```
NaN function div(uint256 a, uint256 b) internal pure returns (uint256) {
NaN   // assert(b > 0); // Solidity automatically throws when dividing by 0
NaN   // uint256 c = a / b;
NaN   // assert(a == b * c + a % b); // There is no case in which this doesn't hold
NaN   return a / b;
NaN }
```

The code meets the specification

Detail for Request 2: SafeMath_sub

08, Sep 2018

Posted by CTK report generator

1ms

Line 59-67 in File CelerToken.sol

```
NaN /*@CTK SafeMath_sub
NaN   @tag spec
NaN   @post __reverted == __has_assertion_failure
NaN   @post __has_overflow == true -> __has_assertion_failure == true
NaN   @post __reverted == false -> __return == a - b
NaN   @post msg == msg__post
NaN   @post (a < b) == __has_assertion_failure
NaN   @post __addr_map == __addr_map__post
NaN */
```

Line 68-71 in File CelerToken.sol

```
NaN function sub(uint256 a, uint256 b) internal pure returns (uint256) {
NaN   assert(b <= a);
NaN   return a - b;
NaN }
```

The code meets the specification

Detail for Request 3: SafeMath_add

08, Sep 2018

Posted by CTK report generator

3ms

Line 76-84 in File CelerToken.sol

```
NaN /*@CTK SafeMath_add
NaN   @tag spec
NaN   @post __reverted == __has_assertion_failure
NaN   @post __has_assertion_failure == __has_overflow
NaN   @post __reverted == false -> c == a + b
NaN   @post msg == msg__post
NaN   @post (a + b < a)  == __has_assertion_failure
NaN   @post __addr_map == __addr_map__post
NaN */
```

Line 85-89 in File CelerToken.sol

```
NaN function add(uint256 a, uint256 b) internal pure returns (uint256 c) {
NaN   c = a + b;
NaN   assert(c >= a);
NaN   return c;
NaN }
```

The code meets the specification

Detail for Request 4: If method completes, integer overflow would not happen.

08, Sep 2018

Posted by CTK report generator

0.5ms

Line 140 in File CelerToken.sol

```
NaN  //@CTK NO_OVERFLOW
```

Line 146-148 in File CelerToken.sol

```
NaN  function totalSupply() public view returns (uint256) {
NaN    return totalSupply_;
NaN  }
```

The code meets the specification

Detail for Request 5: Method will not encounter an assertion failure.

08, Sep 2018

Posted by CTK report generator

0.4ms

Line 141 in File CelerToken.sol

```
NaN  //@CTK NO_ASF
```

Line 146-148 in File CelerToken.sol

```
NaN  function totalSupply() public view returns (uint256) {
NaN    return totalSupply_;
NaN  }
```

The code meets the specification

Detail for Request 6: Buffer overflow / array index out of bound would never happen.

08, Sep 2018

0.3ms

Posted by CTK report generator

Line 142 in File CelerToken.sol

```
NaN //@CTK NO_BUF_OVERFLOW
```

Line 146-148 in File CelerToken.sol

```
NaN function totalSupply() public view returns (uint256) {
NaN   return totalSupply_;
NaN }
```

The code meets the specification

Detail for Request 7: token_total_supply

08, Sep 2018

0.4ms

Posted by CTK report generator

Line 143-145 in File CelerToken.sol

```
NaN /*@CTK token_total_supply
NaN   @post __return == this.totalSupply_
NaN */
```

Line 146-148 in File CelerToken.sol

```
NaN function totalSupply() public view returns (uint256) {
NaN   return totalSupply_;
NaN }
```

The code meets the specification

Detail for Request 8: If method completes, integer overflow would not happen.

Posted by CTK report generator

08, Sep 2018
9.2ms

Line 155 in File CelerToken.sol

```
NaN //@CTK NO_OVERFLOW
```

Line 168-176 in File CelerToken.sol

```
NaN function transfer(address _to, uint256 _value) public returns (bool) {
NaN   require(_to != address(0));
NaN   require(_value <= balances[msg.sender]);
NaN
NaN   balances[msg.sender] = balances[msg.sender].sub(_value);
NaN   balances[_to] = balances[_to].add(_value);
NaN   emit Transfer(msg.sender, _to, _value);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 9: transfer success case

08, Sep 2018

Posted by CTK report generator

102.9ms

Line 156-163 in File CelerToken.sol

```
NaN /*@CTK "transfer success case"
NaN   @tag assume_completion
NaN   @pre _to != address(0)
NaN   @pre balances[msg.sender] >= _value
NaN   @post _to != msg.sender -> __post.balances[msg.sender] == balances[msg.sender] -
NaN   @post _to != msg.sender -> __post.balances[_to] == balances[_to] + _value
NaN   @post __return == true
NaN */
```

Line 168-176 in File CelerToken.sol

```
NaN function transfer(address _to, uint256 _value) public returns (bool) {
NaN   require(_to != address(0));
NaN   require(_value <= balances[msg.sender]);
NaN
NaN   balances[msg.sender] = balances[msg.sender].sub(_value);
NaN   balances[_to] = balances[_to].add(_value);
NaN   emit Transfer(msg.sender, _to, _value);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 10: transfer reverted case

08, Sep 2018

Posted by CTK report generator

7.3ms

Line 164-167 in File CelerToken.sol

```
NaN /*@CTK "transfer reverted case"
NaN   @pre _to == address(0) \/ balances[msg.sender] < _value
NaN   @post __reverted == true
NaN */
```

Line 168-176 in File CelerToken.sol

```
NaN function transfer(address _to, uint256 _value) public returns (bool) {
NaN   require(_to != address(0));
NaN   require(_value <= balances[msg.sender]);
NaN
NaN   balances[msg.sender] = balances[msg.sender].sub(_value);
NaN   balances[_to] = balances[_to].add(_value);
NaN   emit Transfer(msg.sender, _to, _value);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 11: transfer success case

08, Sep 2018

Posted by CTK report generator

0.2ms

Line 183-185 in File CelerToken.sol

```
NaN /*@CTK "transfer success case"
NaN   @post __return == this.balances[_owner]
NaN */
```

Line 186-188 in File CelerToken.sol

```
NaN function balanceOf(address _owner) public view returns (uint256) {
NaN   return balances[_owner];
NaN }
```

The code meets the specification

Detail for Request 12: Ownable constructor

08, Sep 2018

0.5ms

Posted by CTK report generator

Line 213-215 in File CelerToken.sol

```
NaN /*@CTK "Ownable constructor"
NaN   @post post(this).owner == msg.sender
NaN */
```

Line 216-218 in File CelerToken.sol

```
NaN constructor() public {
NaN   owner = msg.sender;
NaN }
```

The code meets the specification

Detail for Request 13: renounceOwnership

08, Sep 2018

1ms

Posted by CTK report generator

Line 237-240 in File CelerToken.sol

```
NaN /*@CTK "renounceOwnership"
NaN   @tag assume_completion
NaN   @post post(this).owner == address(0)
NaN */
```

Line 241-244 in File CelerToken.sol

```
NaN function renounceOwnership() public onlyOwner {
NaN   emit OwnershipRenounced(owner);
NaN   owner = address(0);
NaN }
```

The code meets the specification

Detail for Request 14: onlyOwner_renounceOwnership

08, Sep 2018

Posted by CTK report generator

0.7ms

Line 223-225 in File CelerToken.sol

```
NaN /*@CTK "onlyOwner"
NaN   @post msg.sender != owner -> __reverted
NaN */
```

Line 241-244 in File CelerToken.sol

```
NaN function renounceOwnership() public onlyOwner {
NaN   emit OwnershipRenounced(owner);
NaN   owner = address(0);
NaN }
```

The code meets the specification

Detail for Request 15: transferOwnership

08, Sep 2018

Posted by CTK report generator

3.5ms

Line 250-253 in File CelerToken.sol

```
NaN /*@CTK "transferOwnership"
NaN   @tag assume_completion
NaN   @post post(this).owner == _newOwner
NaN */
```

Line 254-256 in File CelerToken.sol

```
NaN function transferOwnership(address _newOwner) public onlyOwner {
NaN   _transferOwnership(_newOwner);
NaN }
```

The code meets the specification

Detail for Request 16: onlyOwner_transferOwnership

08, Sep 2018

Posted by CTK report generator

1.1ms

Line 223-225 in File CelerToken.sol

```
NaN /*@CTK "onlyOwner"
NaN   @post msg.sender != owner -> __reverted
NaN */
```

Line 254-256 in File CelerToken.sol

```
NaN function transferOwnership(address _newOwner) public onlyOwner {
NaN   _transferOwnership(_newOwner);
NaN }
```

The code meets the specification

Detail for Request 17: _transferOwnership success case

08, Sep 2018

Posted by CTK report generator

2.7ms

Line 262-265 in File CelerToken.sol

```
NaN /*@CTK "_transferOwnership success case"
NaN   @pre _newOwner != address(0)
NaN   @post post(this).owner == _newOwner
NaN */
```

Line 270-274 in File CelerToken.sol

```
NaN function _transferOwnership(address _newOwner) internal {
NaN   require(_newOwner != address(0));
NaN   emit OwnershipTransferred(owner, _newOwner);
NaN   owner = _newOwner;
NaN }
```

The code meets the specification

Detail for Request 18: _transferOwnership reverted case

08, Sep 2018

Posted by CTK report generator

0.6ms

Line 266-269 in File CelerToken.sol

```
NaN /*@CTK "_transferOwnership reverted case"
```

```
NaN   @pre _newOwner == address(0)
NaN   @post __reverted
NaN */
```

Line 270-274 in File CelerToken.sol

```
NaN function _transferOwnership(address _newOwner) internal {
NaN   require(_newOwner != address(0));
NaN   emit OwnershipTransferred(owner, _newOwner);
NaN   owner = _newOwner;
NaN }
```

The code meets the specification

Detail for Request 19: pause

08, Sep 2018

Posted by CTK report generator

4.2ms

Line 314-317 in File CelerToken.sol

```
NaN /*@CTK "pause"
NaN   @tag assume_completion
NaN   @post post(this).paused == true
NaN */
```

Line 318-321 in File CelerToken.sol

```
NaN function pause() onlyOwner whenNotPaused public {
NaN   paused = true;
NaN   emit Pause();
NaN }
```

The code meets the specification

Detail for Request 20: onlyOwner_pause

08, Sep 2018

Posted by CTK report generator

2.6ms

Line 223-225 in File CelerToken.sol

```
NaN /*@CTK "onlyOwner"
NaN   @post msg.sender != owner -> __reverted
NaN */
```

Line 318-321 in File CelerToken.sol

```
NaN function pause() onlyOwner whenNotPaused public {
NaN   paused = true;
NaN   emit Pause();
NaN }
```

The code meets the specification

Detail for Request 21: whenNotPaused_pause

08, Sep 2018

0.8ms

Posted by CTK report generator

Line 292-294 in File CelerToken.sol

```
NaN /*@CTK "whenNotPaused"
NaN   @post this.paused -> __reverted
NaN */
```

Line 318-321 in File CelerToken.sol

```
NaN function pause() onlyOwner whenNotPaused public {
NaN   paused = true;
NaN   emit Pause();
NaN }
```

The code meets the specification

Detail for Request 22: unpause

08, Sep 2018

2.1ms

Posted by CTK report generator

Line 326-329 in File CelerToken.sol

```
NaN /*@CTK "unpause"
NaN   @tag assume_completion
NaN   @post post(this).paused == false
NaN */
```

Line 330-333 in File CelerToken.sol

```
NaN function unpause() onlyOwner whenPaused public {
NaN   paused = false;
```

```
NaN   emit Unpause();
NaN }
```

The code meets the specification

Detail for Request 23: onlyOwner_unpause

08, Sep 2018

Posted by CTK report generator

1.7ms

Line 223-225 in File CelerToken.sol

```
NaN /*@CTK "onlyOwner"
NaN   @post msg.sender != owner -> __reverted
NaN */
```

Line 330-333 in File CelerToken.sol

```
NaN function unpause() onlyOwner whenPaused public {
NaN   paused = false;
NaN   emit Unpause();
NaN }
```

The code meets the specification

Detail for Request 24: whenPaused_unpause

08, Sep 2018

Posted by CTK report generator

0.8ms

Line 303-305 in File CelerToken.sol

```
NaN /*@CTK "whenPaused"
NaN   @post !this.paused -> __reverted
NaN */
```

Line 330-333 in File CelerToken.sol

```
NaN function unpause() onlyOwner whenPaused public {
NaN   paused = false;
NaN   emit Unpause();
NaN }
```

The code meets the specification

Detail for Request 25: transferFrom success

08, Sep 2018

Posted by CTK report generator

169.3ms

Line 355-364 in File CelerToken.sol

```
NaN /*@CTK "transferFrom success"
NaN   @tag assume_completion
NaN   @pre _to != address(0)
NaN   @pre _value <= balances[_from]
NaN   @pre _value <= allowed[_from][msg.sender]
NaN   @post _from != _to -> __post.balances[_from] == balances[_from] - _value
NaN   @post _from != _to -> __post.balances[_to] == balances[_to] + _value
NaN   @post __post.allowed[_from][msg.sender] == allowed[_from][msg.sender] - _value
NaN   @post __return == true
NaN */
```

Line 377-394 in File CelerToken.sol

```
NaN function transferFrom(
NaN   address _from,
NaN   address _to,
NaN   uint256 _value
NaN )
NaN   public
NaN   returns (bool)
NaN {
NaN   require(_to != address(0));
NaN   require(_value <= balances[_from]);
NaN   require(_value <= allowed[_from][msg.sender]);
NaN
NaN   balances[_from] = balances[_from].sub(_value);
NaN   balances[_to] = balances[_to].add(_value);
NaN   allowed[_from][msg.sender] = allowed[_from][msg.sender].sub(_value);
NaN   emit Transfer(_from, _to, _value);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 26: transferFrom failure case 1: no enough balance

08, Sep 2018

Posted by CTK report generator

16ms

Line 365-368 in File CelerToken.sol

```
NaN /*@CTK "transferFrom failure case 1: no enough balance"
NaN   @pre balances[_from] < _value
NaN   @post __reverted
NaN */
```

Line 377-394 in File CelerToken.sol

```
NaN function transferFrom(
NaN   address _from,
NaN   address _to,
NaN   uint256 _value
NaN )
NaN   public
NaN   returns (bool)
NaN {
NaN   require(_to != address(0));
NaN   require(_value <= balances[_from]);
NaN   require(_value <= allowed[_from][msg.sender]);
NaN
NaN   balances[_from] = balances[_from].sub(_value);
NaN   balances[_to] = balances[_to].add(_value);
NaN   allowed[_from][msg.sender] = allowed[_from][msg.sender].sub(_value);
NaN   emit Transfer(_from, _to, _value);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 27: transferFrom failure case 2: no enough allowance

08, Sep 2018

Posted by CTK report generator

17.9ms

Line 369-372 in File CelerToken.sol

```
NaN /*@CTK "transferFrom failure case 2: no enough allowance"
NaN   @pre allowed[_from][msg.sender] < _value
NaN   @post __reverted
NaN */
```

Line 377-394 in File CelerToken.sol

```
NaN function transferFrom(
NaN   address _from,
NaN   address _to,
NaN   uint256 _value
NaN )
NaN   public
NaN   returns (bool)
NaN {
NaN   require(_to != address(0));
NaN   require(_value <= balances[_from]);
NaN   require(_value <= allowed[_from][msg.sender]);
NaN
NaN   balances[_from] = balances[_from].sub(_value);
```

```
NaN   balances[_to] = balances[_to].add(_value);
NaN   allowed[_from][msg.sender] = allowed[_from][msg.sender].sub(_value);
NaN   emit Transfer(_from, _to, _value);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 28: transferFrom failure case 3: _to is 0

08, Sep 2018

Posted by CTK report generator

0.8ms

Line 373-376 in File CelerToken.sol

```
NaN /*@CTK "transferFrom failure case 3: _to is 0"
NaN   @pre _to == address(0)
NaN   @post __reverted
NaN */
```

Line 377-394 in File CelerToken.sol

```
NaN function transferFrom(
NaN   address _from,
NaN   address _to,
NaN   uint256 _value
NaN )
NaN   public
NaN   returns (bool)
NaN {
NaN   require(_to != address(0));
NaN   require(_value <= balances[_from]);
NaN   require(_value <= allowed[_from][msg.sender]);
NaN
NaN   balances[_from] = balances[_from].sub(_value);
NaN   balances[_to] = balances[_to].add(_value);
NaN   allowed[_from][msg.sender] = allowed[_from][msg.sender].sub(_value);
NaN   emit Transfer(_from, _to, _value);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 29: If method completes, integer overflow would not happen.

08, Sep 2018

0.5ms

Posted by CTK report generator

Line 406 in File CelerToken.sol

```
NaN //@CTK NO_OVERFLOW
```

Line 412-416 in File CelerToken.sol

```
NaN function approve(address _spender, uint256 _value) public returns (bool) {
NaN   allowed[msg.sender][_spender] = _value;
NaN   emit Approval(msg.sender, _spender, _value);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 30: Method will not encounter an assertion failure.

08, Sep 2018

0.4ms

Posted by CTK report generator

Line 407 in File CelerToken.sol

```
NaN //@CTK NO_ASF
```

Line 412-416 in File CelerToken.sol

```
NaN function approve(address _spender, uint256 _value) public returns (bool) {
NaN   allowed[msg.sender][_spender] = _value;
NaN   emit Approval(msg.sender, _spender, _value);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 31: approve transfer allowance

08, Sep 2018

1.1ms

Posted by CTK report generator

Line 408-411 in File CelerToken.sol

```
NaN /*@CTK "approve transfer allowance"
NaN   @post post(this).allowed[msg.sender][_spender] == _value
NaN   @post __return == true
NaN */
```

Line 412-416 in File CelerToken.sol

```
NaN function approve(address _spender, uint256 _value) public returns (bool) {
NaN   allowed[msg.sender][_spender] = _value;
NaN   emit Approval(msg.sender, _spender, _value);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 32: get the allowance

08, Sep 2018

Posted by CTK report generator

0.4ms

Line 424-427 in File CelerToken.sol

```
NaN /*@CTK "get the allowance"
NaN   @post __return == allowed[_owner][_spender]
NaN   @post this == post(this)
NaN */
```

Line 428-437 in File CelerToken.sol

```
NaN function allowance(
NaN   address _owner,
NaN   address _spender
NaN  )
NaN   public
NaN   view
NaN   returns (uint256)
NaN {
NaN   return allowed[_owner][_spender];
NaN }
```

The code meets the specification

Detail for Request 33: increaseApproval ok

08, Sep 2018

Posted by CTK report generator

1.6ms

Line 448-452 in File CelerToken.sol

```
NaN /*@CTK "increaseApproval ok"
NaN   @tag assume_completion
NaN   @post post(this).allowed[msg.sender][_spender] == this.allowed[msg.sender][_spenc
NaN   @post __return == true
NaN */
```

Line 453-464 in File CelerToken.sol

```
NaN function increaseApproval(
NaN   address _spender,
NaN   uint256 _addedValue
NaN )
NaN   public
NaN   returns (bool)
NaN {
NaN   allowed[msg.sender][_spender] = (
NaN     allowed[msg.sender][_spender].add(_addedValue));
NaN   emit Approval(msg.sender, _spender, allowed[msg.sender][_spender]);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 34: decreaseApproval case if

08, Sep 2018

Posted by CTK report generator

1.9ms

Line 475-480 in File CelerToken.sol

```
NaN /*@CTK "decreaseApproval case if"
NaN   @tag assume_completion
NaN   @pre _subtractedValue > allowed[msg.sender][_spender]
NaN   @post post(this).allowed[msg.sender][_spender] == 0
NaN   @post __return == true
NaN */
```

Line 487-502 in File CelerToken.sol

```
NaN function decreaseApproval(
NaN   address _spender,
NaN   uint256 _subtractedValue
NaN )
NaN   public
NaN   returns (bool)
NaN {
NaN   uint256 oldValue = allowed[msg.sender][_spender];
NaN   if (_subtractedValue > oldValue) {
NaN     allowed[msg.sender][_spender] = 0;
NaN   } else {
NaN     allowed[msg.sender][_spender] = oldValue.sub(_subtractedValue);
NaN   }
NaN   emit Approval(msg.sender, _spender, allowed[msg.sender][_spender]);
```

```
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 35: decreaseApproval case else

08, Sep 2018

Posted by CTK report generator

1.9ms

Line 481-486 in File CelerToken.sol

```
NaN /*@CTK "decreaseApproval case else"
NaN   @tag assume_completion
NaN   @pre _subtractedValue <= allowed[msg.sender][_spender]
NaN   @post post(this).allowed[msg.sender][_spender] == this.allowed[msg.sender][_spenc
NaN   @post __return == true
NaN */
```

Line 487-502 in File CelerToken.sol

```
NaN function decreaseApproval(
NaN   address _spender,
NaN   uint256 _subtractedValue
NaN )
NaN   public
NaN   returns (bool)
NaN {
NaN   uint256 oldValue = allowed[msg.sender][_spender];
NaN   if (_subtractedValue > oldValue) {
NaN     allowed[msg.sender][_spender] = 0;
NaN   } else {
NaN     allowed[msg.sender][_spender] = oldValue.sub(_subtractedValue);
NaN   }
NaN   emit Approval(msg.sender, _spender, allowed[msg.sender][_spender]);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 36: addAddressToWhitelist

08, Sep 2018

Posted by CTK report generator

1.3ms

Line 537-541 in File CelerToken.sol

```
NaN  /*@CTK "addAddressToWhitelist"
NaN    @tag assume_completion
NaN    @post post(this).whitelisted[_operator] == true
NaN    @post __return == true
NaN  */
```

Line 542-550 in File CelerToken.sol

```
NaN  function addAddressToWhitelist(address _operator)
NaN    onlyOwner
NaN    public
NaN    returns (bool)
NaN  {
NaN    whitelisted[_operator] = true;
NaN    emit WhitelistAdded(_operator);
NaN    return true;
NaN  }
```

The code meets the specification

Detail for Request 37: onlyOwner_addAddressToWhitelist

08, Sep 2018

Posted by CTK report generator

0.7ms

Line 223-225 in File CelerToken.sol

```
NaN  /*@CTK "onlyOwner"
NaN    @post msg.sender != owner -> __reverted
NaN  */
```

Line 542-550 in File CelerToken.sol

```
NaN  function addAddressToWhitelist(address _operator)
NaN    onlyOwner
NaN    public
NaN    returns (bool)
NaN  {
NaN    whitelisted[_operator] = true;
NaN    emit WhitelistAdded(_operator);
NaN    return true;
NaN  }
```

The code meets the specification

Detail for Request 38: whitelist happy case

08, Sep 2018

Posted by CTK report generator

0.3ms

Line 555-558 in File CelerToken.sol

```
NaN /*@CTK "whitelist happy case"
NaN   @post __return == this.whitelisted[_operator]
NaN   @post post(this) == this
NaN */
```

Line 559-566 in File CelerToken.sol

```
NaN function whitelist(address _operator)
NaN   public
NaN   view
NaN   returns (bool)
NaN {
NaN   bool result = whitelisted[_operator];
NaN   return result;
NaN }
```

The code meets the specification

Detail for Request 39: addAddressesToWhitelist happy case

08, Sep 2018

Posted by CTK report generator

5.9ms

Line 574-578 in File CelerToken.sol

```
NaN /*@CTK "addAddressesToWhitelist happy case"
NaN   @tag assume_completion
NaN   @post forall i: uint. (i >= 0 /\ i < _operators.length) -> post(this).whitelistec
NaN   @post __return == true
NaN */
```

Line 579-595 in File CelerToken.sol

```
NaN function addAddressesToWhitelist(address[] _operators)
NaN   onlyOwner
NaN   public
NaN   returns (bool)
NaN {
NaN   /*@CTK addAddressesToWhitelist_forloop
NaN     @inv i <= _operators.length
NaN     @inv _operators == _operators__pre
NaN     @inv forall j: uint. (j >= 0 /\ j < i) -> this.whitelisted[_operators[j]] == tr
NaN     @post i == _operators.length
NaN     @post !__should_return
NaN   */
NaN   for (uint256 i = 0; i < _operators.length; i++) {
```

```
NaN     require(addAddressToWhitelist(_operators[i]));
NaN   }
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 40: onlyOwner_addAddressesToWhitelist

08, Sep 2018

Posted by CTK report generator

1ms

Line 223-225 in File CelerToken.sol

```
NaN /*@CTK "onlyOwner"
NaN   @post msg.sender != owner -> __reverted
NaN */
```

Line 579-595 in File CelerToken.sol

```
NaN function addAddressesToWhitelist(address[] _operators)
NaN   onlyOwner
NaN   public
NaN   returns (bool)
NaN {
NaN   /*@CTK addAddressesToWhitelist_forloop
NaN     @inv i <= _operators.length
NaN     @inv _operators == _operators__pre
NaN     @inv forall j: uint. (j >= 0 /\ j < i) -> this.whitelisted[_operators[j]] == tr
NaN     @post i == _operators.length
NaN     @post !__should_return
NaN   */
NaN   for (uint256 i = 0; i < _operators.length; i++) {
NaN     require(addAddressToWhitelist(_operators[i]));
NaN   }
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 41: removeAddressFromWhitelist happy case

08, Sep 2018

Posted by CTK report generator

1.3ms

Line 603-607 in File CelerToken.sol

```
NaN /*@CTK "removeAddressFromWhitelist happy case"
NaN   @tag assume_completion
NaN   @post post(this).whitelisted[_operator] == false
NaN   @post __return == true
NaN */
```

Line 608-616 in File CelerToken.sol

```
NaN function removeAddressFromWhitelist(address _operator)
NaN   onlyOwner
NaN   public
NaN   returns (bool)
NaN {
NaN   whitelisted[_operator] = false;
NaN   emit WhitelistRemoved(_operator);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 42: onlyOwner_removeAddressFromWhitelist

08, Sep 2018

Posted by CTK report generator

0.7ms

Line 223-225 in File CelerToken.sol

```
NaN /*@CTK "onlyOwner"
NaN   @post msg.sender != owner -> __reverted
NaN */
```

Line 608-616 in File CelerToken.sol

```
NaN function removeAddressFromWhitelist(address _operator)
NaN   onlyOwner
NaN   public
NaN   returns (bool)
NaN {
NaN   whitelisted[_operator] = false;
NaN   emit WhitelistRemoved(_operator);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 43: removeAddressesFromWhitelist happy case

08, Sep 2018

Posted by CTK report generator

4.5ms

Line 624-628 in File CelerToken.sol

```
NaN oveAddressesFromWhitelist happy case"
NaN e_completion
NaN ll i: uint. (i >= 0 /\ i < _operators.length) -> post(this).whitelisted[_operators[
NaN turn == true
NaN
```

Line 629-645 in File CelerToken.sol

```
NaN function removeAddressesFromWhitelist(address[] _operators)
NaN   onlyOwner
NaN   public
NaN   returns (bool)
NaN {
NaN   /*@CTK removeAddressesFromWhitelist_forLoop
NaN     @inv i <= _operators.length
NaN     @inv _operators == _operators__pre
NaN     @inv forall j: uint. (j >= 0 /\ j < i) -> this.whitelisted[_operators[j]] == fa
NaN     @post i == _operators.length
NaN     @post !__should_return
NaN   */
NaN   for (uint256 i = 0; i < _operators.length; i++) {
NaN     require(removeAddressFromWhitelist(_operators[i]));
NaN   }
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 44: onlyOwner_removeAddressesFromWhitelist

08, Sep 2018

Posted by CTK report generator

1.2ms

Line 223-225 in File CelerToken.sol

```
NaN /*@CTK "onlyOwner"
NaN   @post msg.sender != owner -> __reverted
NaN */
```

Line 629-645 in File CelerToken.sol

```
NaN function removeAddressesFromWhitelist(address[] _operators)
NaN   onlyOwner
NaN   public
NaN   returns (bool)
```

```
NaN {
NaN   /*@CTK removeAddressesFromWhitelist_forLoop
NaN     @inv i <= _operators.length
NaN     @inv _operators == _operators__pre
NaN     @inv forall j: uint. (j >= 0 /\ j < i) -> this.whitelisted[_operators[j]] == fa
NaN     @post i == _operators.length
NaN     @post !__should_return
NaN   */
NaN   for (uint256 i = 0; i < _operators.length; i++) {
NaN     require(removeAddressFromWhitelist(_operators[i]));
NaN   }
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 45: PausableToken transfer success case

08, Sep 2018

Posted by CTK report generator

125.3ms

Line 656-661 in File CelerToken.sol

```
NaN /*@CTK "PausableToken transfer success case"
NaN   @tag assume_completion
NaN   @post _to != msg.sender -> __post.balances[msg.sender] == balances[msg.sender] -
NaN   @post _to != msg.sender -> __post.balances[_to] == balances[_to] + _value
NaN   @post __return == true
NaN */
```

Line 662-671 in File CelerToken.sol

```
NaN function transfer(
NaN   address _to,
NaN   uint256 _value
NaN )
NaN   public
NaN   whenNotPaused
NaN   returns (bool)
NaN {
NaN   return super.transfer(_to, _value);
NaN }
```

The code meets the specification

Detail for Request 46: whenNotPaused_transfer

08, Sep 2018

1.3ms

Posted by CTK report generator

Line 292-294 in File CelerToken.sol

```
NaN /*@CTK "whenNotPaused"
NaN   @post this.paused -> __reverted
NaN */
```

Line 662-671 in File CelerToken.sol

```
NaN function transfer(
NaN   address _to,
NaN   uint256 _value
NaN )
NaN   public
NaN   whenNotPaused
NaN   returns (bool)
NaN {
NaN   return super.transfer(_to, _value);
NaN }
```

The code meets the specification

Detail for Request 47: PausableToken transferFrom success

08, Sep 2018

293.9ms

Posted by CTK report generator

Line 673-679 in File CelerToken.sol

```
NaN /*@CTK "PausableToken transferFrom success"
NaN   @tag assume_completion
NaN   @post _from != _to -> __post.balances[_from] == balances[_from] - _value
NaN   @post _from != _to -> __post.balances[_to] == balances[_to] + _value
NaN   @post __post.allowed[_from][msg.sender] == allowed[_from][msg.sender] - _value
NaN   @post __return == true
NaN */
```

Line 680-690 in File CelerToken.sol

```
NaN function transferFrom(
NaN   address _from,
NaN   address _to,
NaN   uint256 _value
NaN )
NaN   public
NaN   whenNotPaused
NaN   returns (bool)
NaN {
NaN   return super.transferFrom(_from, _to, _value);
NaN }
```

The code meets the specification

Detail for Request 48: whenNotPaused_transferFrom

08, Sep 2018

Posted by CTK report generator

1.6ms

Line 292-294 in File CelerToken.sol

```
NaN /*@CTK "whenNotPaused"
NaN   @post this.paused -> __reverted
NaN */
```

Line 680-690 in File CelerToken.sol

```
NaN function transferFrom(
NaN   address _from,
NaN   address _to,
NaN   uint256 _value
NaN )
NaN   public
NaN   whenNotPaused
NaN   returns (bool)
NaN {
NaN   return super.transferFrom(_from, _to, _value);
NaN }
```

The code meets the specification

Detail for Request 49: PausableToken approve transfer allowance

08, Sep 2018

Posted by CTK report generator

2.4ms

Line 692-696 in File CelerToken.sol

```
NaN /*@CTK "PausableToken approve transfer allowance"
NaN   @pre paused == false
NaN   @post post(this).allowed[msg.sender][_spender] == _value
NaN   @post __return == true
NaN */
```

Line 697-706 in File CelerToken.sol

```
NaN function approve(
NaN   address _spender,
NaN   uint256 _value
NaN )
```

```
NaN   public
NaN   whenNotPaused
NaN   returns (bool)
NaN {
NaN   return super.approve(_spender, _value);
NaN }
```

The code meets the specification

Detail for Request 50: whenNotPaused_approve

08, Sep 2018

Posted by CTK report generator

0.9ms

Line 292-294 in File CelerToken.sol

```
NaN /*@CTK "whenNotPaused"
NaN   @post this.paused -> __reverted
NaN */
```

Line 697-706 in File CelerToken.sol

```
NaN function approve(
NaN   address _spender,
NaN   uint256 _value
NaN )
NaN   public
NaN   whenNotPaused
NaN   returns (bool)
NaN {
NaN   return super.approve(_spender, _value);
NaN }
```

The code meets the specification

Detail for Request 51: increaseApproval ok

08, Sep 2018

Posted by CTK report generator

3ms

Line 707-710 in File CelerToken.sol

```
NaN /*@CTK "increaseApproval ok"
NaN   @tag assume_completion
NaN   @post post(this).allowed[msg.sender][_spender] == this.allowed[msg.sender][_spenc
```

```
NaN */
```

Line 711-720 in File CelerToken.sol

```
NaN function increaseApproval(
NaN   address _spender,
NaN   uint _addedValue
NaN )
NaN   public
NaN   whenNotPaused
NaN   returns (bool success)
NaN {
NaN   return super.increaseApproval(_spender, _addedValue);
NaN }
```

The code meets the specification

Detail for Request 52: whenNotPaused_increaseApproval

08, Sep 2018

Posted by CTK report generator

1ms

Line 292-294 in File CelerToken.sol

```
NaN /*@CTK "whenNotPaused"
NaN   @post this.paused -> __reverted
NaN */
```

Line 711-720 in File CelerToken.sol

```
NaN function increaseApproval(
NaN   address _spender,
NaN   uint _addedValue
NaN )
NaN   public
NaN   whenNotPaused
NaN   returns (bool success)
NaN {
NaN   return super.increaseApproval(_spender, _addedValue);
NaN }
```

The code meets the specification

Detail for Request 53: decreaseApproval case if

08, Sep 2018

Posted by CTK report generator

29.4ms

Line 722-726 in File CelerToken.sol

```
NaN /*@CTK "decreaseApproval case if"
NaN   @tag assume_completion
NaN   @pre _subtractedValue > allowed[msg.sender][_spender]
NaN   @post post(this).allowed[msg.sender][_spender] == 0
NaN */
```

Line 732-741 in File CelerToken.sol

```
NaN function decreaseApproval(
NaN   address _spender,
NaN   uint _subtractedValue
NaN )
NaN   public
NaN   whenNotPaused
NaN   returns (bool success)
NaN {
NaN   return super.decreaseApproval(_spender, _subtractedValue);
NaN }
```

The code meets the specification

Detail for Request 54: decreaseApproval case else

08, Sep 2018

Posted by CTK report generator

48.3ms

Line 727-731 in File CelerToken.sol

```
NaN /*@CTK "decreaseApproval case else"
NaN   @tag assume_completion
NaN   @pre _subtractedValue <= allowed[msg.sender][_spender]
NaN   @post post(this).allowed[msg.sender][_spender] == this.allowed[msg.sender][_spenc
NaN */
```

Line 732-741 in File CelerToken.sol

```
NaN function decreaseApproval(
NaN   address _spender,
NaN   uint _subtractedValue
NaN )
NaN   public
NaN   whenNotPaused
NaN   returns (bool success)
NaN {
NaN   return super.decreaseApproval(_spender, _subtractedValue);
NaN }
```

The code meets the specification

Detail for Request 55: whenNotPaused_decreaseApproval

08, Sep 2018

Posted by CTK report generator

1.2ms

Line 292-294 in File CelerToken.sol

```
NaN /*@CTK "whenNotPaused"
NaN   @post this.paused -> __reverted
NaN */
```

Line 732-741 in File CelerToken.sol

```
NaN function decreaseApproval(
NaN   address _spender,
NaN   uint _subtractedValue
NaN )
NaN   public
NaN   whenNotPaused
NaN   returns (bool success)
NaN {
NaN   return super.decreaseApproval(_spender, _subtractedValue);
NaN }
```

The code meets the specification

Detail for Request 56: CelerToken constructor

08, Sep 2018

Posted by CTK report generator

2.7ms

Line 778-781 in File CelerToken.sol

```
NaN /*@CTK "CelerToken constructor"
NaN   @post post(this).totalSupply_ == INITIAL_SUPPLY
NaN   @post post(this).balances[msg.sender] == INITIAL_SUPPLY
NaN */
```

Line 782-785 in File CelerToken.sol

```
NaN constructor() public {
NaN   totalSupply_ = INITIAL_SUPPLY;
NaN   balances[msg.sender] = INITIAL_SUPPLY;
NaN }
```

The code meets the specification

Detail for Request 57: CelerToken transfer success case

08, Sep 2018

Posted by CTK report generator

240.8ms

Line 791-796 in File CelerToken.sol

```
NaN /*@CTK "CelerToken transfer success case"
NaN   @tag assume_completion
NaN   @post _to != msg.sender -> __post.balances[msg.sender] == balances[msg.sender] -
NaN   @post _to != msg.sender -> __post.balances[_to] == balances[_to] + _value
NaN   @post __return == true
NaN */
```

Line 797-807 in File CelerToken.sol

```
NaN function transfer(
NaN   address _to,
NaN   uint256 _value
NaN )
NaN   public
NaN   onlyIfTransferable
NaN   onlyValidReceiver(_to)
NaN   returns (bool)
NaN {
NaN   return super.transfer(_to, _value);
NaN }
```

The code meets the specification

Detail for Request 58: onlyValidReceiver_transfer

08, Sep 2018

Posted by CTK report generator

7.9ms

Line 760-763 in File CelerToken.sol

```
NaN /*@CTK onlyValidReceiver
NaN   @pre this.transferOpened == false /\ whitelisted[msg.sender] == false /\ msg.senc
NaN   @post __reverted
NaN */
```

Line 797-807 in File CelerToken.sol

```
NaN function transfer(
NaN   address _to,
NaN   uint256 _value
```

```
NaN )
NaN   public
NaN   onlyIfTransferable
NaN   onlyValidReceiver(_to)
NaN   returns (bool)
NaN {
NaN   return super.transfer(_to, _value);
NaN }
```

The code meets the specification

Detail for Request 59: onlyValidReceiver_transfer

08, Sep 2018

Posted by CTK report generator

7.9ms

Line 769-771 in File CelerToken.sol

```
NaN /*@CTK onlyValidReceiver
NaN   @post _to == address(this) -> __reverted
NaN */
```

Line 797-807 in File CelerToken.sol

```
NaN function transfer(
NaN   address _to,
NaN   uint256 _value
NaN )
NaN   public
NaN   onlyIfTransferable
NaN   onlyValidReceiver(_to)
NaN   returns (bool)
NaN {
NaN   return super.transfer(_to, _value);
NaN }
```

The code meets the specification

Detail for Request 60: transferFrom transferFrom success

08, Sep 2018

Posted by CTK report generator

392.5ms

Line 813-819 in File CelerToken.sol

```
NaN /*@CTK "transferFrom transferFrom success"
NaN   @tag assume_completion
NaN   @post _from != _to -> __post.balances[_from] == balances[_from] - _value
NaN   @post _from != _to -> __post.balances[_to] == balances[_to] + _value
NaN   @post __post.allowed[_from][msg.sender] == allowed[_from][msg.sender] - _value
NaN   @post __return == true
NaN */
```

Line 820-831 in File CelerToken.sol

```
NaN function transferFrom(
NaN   address _from,
NaN   address _to,
NaN   uint256 _value
NaN )
NaN   public
NaN   onlyIfTransferable
NaN   onlyValidReceiver(_to)
NaN   returns (bool)
NaN {
NaN   return super.transferFrom(_from, _to, _value);
NaN }
```

The code meets the specification

Detail for Request 61: onlyValidReceiver_transferFrom

08, Sep 2018

Posted by CTK report generator

8.3ms

Line 760-763 in File CelerToken.sol

```
NaN /*@CTK onlyValidReceiver
NaN   @pre this.transferOpened == false /\ whitelisted[msg.sender] == false /\ msg.senc
NaN   @post __reverted
NaN */
```

Line 820-831 in File CelerToken.sol

```
NaN function transferFrom(
NaN   address _from,
NaN   address _to,
NaN   uint256 _value
NaN )
NaN   public
NaN   onlyIfTransferable
NaN   onlyValidReceiver(_to)
NaN   returns (bool)
NaN {
NaN   return super.transferFrom(_from, _to, _value);
NaN }
```

The code meets the specification

Detail for Request 62: onlyValidReceiver_transferFrom

08, Sep 2018

Posted by CTK report generator

8.3ms

Line 769-771 in File CelerToken.sol

```
NaN /*@CTK onlyValidReceiver
NaN   @post _to == address(this) -> __reverted
NaN */
```

Line 820-831 in File CelerToken.sol

```
NaN function transferFrom(
NaN   address _from,
NaN   address _to,
NaN   uint256 _value
NaN )
NaN   public
NaN   onlyIfTransferable
NaN   onlyValidReceiver(_to)
NaN   returns (bool)
NaN {
NaN   return super.transferFrom(_from, _to, _value);
NaN }
```

The code meets the specification

Detail for Request 63: openTransfer ok

08, Sep 2018

Posted by CTK report generator

2.1ms

Line 837-840 in File CelerToken.sol

```
NaN /*@CTK "openTransfer ok"
NaN   @tag assume_completion
NaN   @post post(this).transferOpened == true
NaN */
```

Line 841-843 in File CelerToken.sol

```
NaN function openTransfer() external onlyOwner {
NaN   transferOpened = true;
NaN }
```

The code meets the specification

Detail for Request 64: onlyOwner_openTransfer

08, Sep 2018

Posted by CTK report generator

0.6ms

Line 223-225 in File CelerToken.sol

```
NaN /*@CTK "onlyOwner"
NaN   @post msg.sender != owner -> __reverted
NaN */
```

Line 841-843 in File CelerToken.sol

```
NaN function openTransfer() external onlyOwner {
NaN   transferOpened = true;
NaN }
```

The code meets the specification

CertiK

Certi Request Report

79 out of 79 specs are satisfied.

Detail for Request 0: Ownable constructor

08, Sep 2018

0.5ms

Posted by CTK report generator

Line 379-381 in File CelerTimelock.sol

```
NaN /*@CTK "Ownable constructor"
NaN   @post post(this).owner == msg.sender
NaN */
```

Line 382-384 in File CelerTimelock.sol

```
NaN constructor() public {
NaN   owner = msg.sender;
```

```
NaN }
```

The code meets the specification

Detail for Request 1: renounceOwnership

08, Sep 2018

Posted by CTK report generator

1ms

Line 403-406 in File CelerTimelock.sol

```
NaN /*@CTK "renounceOwnership"
NaN   @tag assume_completion
NaN   @post post(this).owner == address(0)
NaN */
```

Line 407-410 in File CelerTimelock.sol

```
NaN function renounceOwnership() public onlyOwner {
NaN   emit OwnershipRenounced(owner);
NaN   owner = address(0);
NaN }
```

The code meets the specification

Detail for Request 2: onlyOwner_renounceOwnership

08, Sep 2018

Posted by CTK report generator

0.7ms

Line 389-391 in File CelerTimelock.sol

```
NaN /*@CTK "onlyOwner"
NaN   @post msg.sender != owner -> __reverted
NaN */
```

Line 407-410 in File CelerTimelock.sol

```
NaN function renounceOwnership() public onlyOwner {
NaN   emit OwnershipRenounced(owner);
NaN   owner = address(0);
NaN }
```

The code meets the specification

Detail for Request 3: transferOwnership

08, Sep 2018

Posted by CTK report generator

3.5ms

Line 416-419 in File CelerTimelock.sol

```
NaN /*@CTK "transferOwnership"
NaN   @tag assume_completion
NaN   @post post(this).owner == _newOwner
NaN */
```

Line 420-422 in File CelerTimelock.sol

```
NaN function transferOwnership(address _newOwner) public onlyOwner {
NaN   _transferOwnership(_newOwner);
NaN }
```

The code meets the specification

Detail for Request 4: onlyOwner_transferOwnership

08, Sep 2018

Posted by CTK report generator

1.1ms

Line 389-391 in File CelerTimelock.sol

```
NaN /*@CTK "onlyOwner"
NaN   @post msg.sender != owner -> __reverted
NaN */
```

Line 420-422 in File CelerTimelock.sol

```
NaN function transferOwnership(address _newOwner) public onlyOwner {
NaN   _transferOwnership(_newOwner);
NaN }
```

The code meets the specification

Detail for Request 5: _transferOwnership success case

08, Sep 2018

Posted by CTK report generator

2.7ms

Line 428-431 in File CelerTimelock.sol

```
NaN /*@CTK "_transferOwnership success case"
NaN   @pre _newOwner != address(0)
NaN   @post post(this).owner == _newOwner
NaN */
```

Line 436-440 in File CelerTimelock.sol

```
NaN function _transferOwnership(address _newOwner) internal {
NaN   require(_newOwner != address(0));
NaN   emit OwnershipTransferred(owner, _newOwner);
NaN   owner = _newOwner;
NaN }
```

The code meets the specification

Detail for Request 6: _transferOwnership reverted case

08, Sep 2018

Posted by CTK report generator

0.6ms

Line 432-435 in File CelerTimelock.sol

```
NaN /*@CTK "_transferOwnership reverted case"
NaN   @pre _newOwner == address(0)
NaN   @post __reverted
NaN */
```

Line 436-440 in File CelerTimelock.sol

```
NaN function _transferOwnership(address _newOwner) internal {
NaN   require(_newOwner != address(0));
NaN   emit OwnershipTransferred(owner, _newOwner);
NaN   owner = _newOwner;
NaN }
```

The code meets the specification

Detail for Request 7: SafeMath_mul

08, Sep 2018

353.8ms

Posted by CTK report generator

Line 180-189 in File CelerTimelock.sol

```
NaN /*@CTK SafeMath_mul
NaN   @tag spec
NaN   @post __reverted == __has_assertion_failure
NaN   @post __has_assertion_failure == __has_overflow
NaN   @post __reverted == false -> c == a * b
NaN   @post a == 0 -> c == 0
NaN   @post msg == msg__post
NaN   @post (a > 0 && (a * b / a != b)) == __has_assertion_failure
NaN   @post __addr_map == __addr_map__post
NaN */
```

Line 190-201 in File CelerTimelock.sol

```
NaN function mul(uint256 a, uint256 b) internal pure returns (uint256 c) {
NaN   // Gas optimization: this is cheaper than asserting 'a' not being zero, but the
NaN   // benefit is lost if 'b' is also tested.
NaN   // See: https://github.com/OpenZeppelin/openzeppelin-solidity/pull/522
NaN   if (a == 0) {
NaN     return 0;
NaN   }
NaN
NaN   c = a * b;
NaN   assert(c / a == b);
NaN   return c;
NaN }
```

The code meets the specification

Detail for Request 8: SafeMath_div

08, Sep 2018

0.4ms

Posted by CTK report generator

Line 206-214 in File CelerTimelock.sol

```
NaN /*@CTK SafeMath_div
NaN   @tag spec
NaN   @post __reverted == __has_assertion_failure
NaN   @post b == 0 -> __reverted == true  // solidity throws on 0.
NaN   @post __has_overflow == true -> __has_assertion_failure == true
NaN   @post __reverted == false -> __return == a / b
NaN   @post msg == msg__post
NaN   @post __addr_map == __addr_map__post
NaN */
```

Line 215-220 in File CelerTimelock.sol

```
NaN function div(uint256 a, uint256 b) internal pure returns (uint256) {
NaN   // assert(b > 0); // Solidity automatically throws when dividing by 0
NaN   // uint256 c = a / b;
NaN   // assert(a == b * c + a % b); // There is no case in which this doesn't hold
NaN   return a / b;
NaN }
```

The code meets the specification

Detail for Request 9: SafeMath_sub

08, Sep 2018

Posted by CTK report generator

1ms

Line 225-233 in File CelerTimelock.sol

```
NaN /*@CTK SafeMath_sub
NaN   @tag spec
NaN   @post __reverted == __has_assertion_failure
NaN   @post __has_overflow == true -> __has_assertion_failure == true
NaN   @post __reverted == false -> __return == a - b
NaN   @post msg == msg__post
NaN   @post (a < b) == __has_assertion_failure
NaN   @post __addr_map == __addr_map__post
NaN */
```

Line 234-237 in File CelerTimelock.sol

```
NaN function sub(uint256 a, uint256 b) internal pure returns (uint256) {
NaN   assert(b <= a);
NaN   return a - b;
NaN }
```

The code meets the specification

Detail for Request 10: SafeMath_add

08, Sep 2018

Posted by CTK report generator

3ms

Line 242-250 in File CelerTimelock.sol

```
NaN /*@CTK SafeMath_add
NaN   @tag spec
```

```
NaN   @post __reverted == __has_assertion_failure
NaN   @post __has_assertion_failure == __has_overflow
NaN   @post __reverted == false -> c == a + b
NaN   @post msg == msg__post
NaN   @post (a + b < a)  == __has_assertion_failure
NaN   @post __addr_map == __addr_map__post
NaN */
```

Line 251-255 in File CelerTimelock.sol

```
NaN function add(uint256 a, uint256 b) internal pure returns (uint256 c) {
NaN   c = a + b;
NaN   assert(c >= a);
NaN   return c;
NaN }
```

The code meets the specification

Detail for Request 11: If method completes, integer overflow would not happen.

08, Sep 2018

Posted by CTK report generator

0.8ms

Line 306 in File CelerTimelock.sol

```
NaN //@CTK NO_OVERFLOW
```

Line 312-314 in File CelerTimelock.sol

```
NaN function totalSupply() public view returns (uint256) {
NaN   return totalSupply_;
NaN }
```

The code meets the specification

Detail for Request 12: Method will not encounter an assertion failure.

08, Sep 2018

Posted by CTK report generator

0.5ms

Line 307 in File CelerTimelock.sol

```
NaN //@CTK NO_ASF
```

Line 312-314 in File CelerTimelock.sol

```
NaN function totalSupply() public view returns (uint256) {
NaN   return totalSupply_;
NaN }
```

The code meets the specification

Detail for Request 13: Buffer overflow / array index out of bound would never happen.

08, Sep 2018

Posted by CTK report generator

0.6ms

Line 308 in File CelerTimelock.sol

```
NaN //@CTK NO_BUF_OVERFLOW
```

Line 312-314 in File CelerTimelock.sol

```
NaN function totalSupply() public view returns (uint256) {
NaN   return totalSupply_;
NaN }
```

The code meets the specification

Detail for Request 14: token_total_supply

08, Sep 2018

Posted by CTK report generator

0.2ms

Line 309-311 in File CelerTimelock.sol

```
NaN /*@CTK token_total_supply
NaN   @post __return == this.totalSupply_
NaN */
```

Line 312-314 in File CelerTimelock.sol

```
NaN function totalSupply() public view returns (uint256) {
NaN   return totalSupply_;
NaN }
```

The code meets the specification

Detail for Request 15: If method completes, integer overflow would not happen.

08, Sep 2018

Posted by CTK report generator

12.5ms

Line 321 in File CelerTimelock.sol

```
NaN //@CTK NO_OVERFLOW
```

Line 334-342 in File CelerTimelock.sol

```
NaN function transfer(address _to, uint256 _value) public returns (bool) {
NaN   require(_to != address(0));
NaN   require(_value <= balances[msg.sender]);
NaN
NaN   balances[msg.sender] = balances[msg.sender].sub(_value);
NaN   balances[_to] = balances[_to].add(_value);
NaN   emit Transfer(msg.sender, _to, _value);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 16: transfer success case

08, Sep 2018

Posted by CTK report generator

153.3ms

Line 322-329 in File CelerTimelock.sol

```
NaN /*@CTK "transfer success case"
NaN   @tag assume_completion
NaN   @pre _to != address(0)
NaN   @pre balances[msg.sender] >= _value
NaN   @post _to != msg.sender -> __post.balances[msg.sender] == balances[msg.sender] -
NaN   @post _to != msg.sender -> __post.balances[_to] == balances[_to] + _value
NaN   @post __return == true
NaN */
```

Line 334-342 in File CelerTimelock.sol

```
NaN function transfer(address _to, uint256 _value) public returns (bool) {
NaN   require(_to != address(0));
```

```
NaN   require(_value <= balances[msg.sender]);
NaN
NaN   balances[msg.sender] = balances[msg.sender].sub(_value);
NaN   balances[_to] = balances[_to].add(_value);
NaN   emit Transfer(msg.sender, _to, _value);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 17: transfer reverted case

08, Sep 2018

Posted by CTK report generator

14.4ms

Line 330-333 in File CelerTimelock.sol

```
NaN /*@CTK "transfer reverted case"
NaN   @pre _to == address(0) \/ balances[msg.sender] < _value
NaN   @post __reverted == true
NaN */
```

Line 334-342 in File CelerTimelock.sol

```
NaN function transfer(address _to, uint256 _value) public returns (bool) {
NaN   require(_to != address(0));
NaN   require(_value <= balances[msg.sender]);
NaN
NaN   balances[msg.sender] = balances[msg.sender].sub(_value);
NaN   balances[_to] = balances[_to].add(_value);
NaN   emit Transfer(msg.sender, _to, _value);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 18: transfer success case

08, Sep 2018

Posted by CTK report generator

0.3ms

Line 349-351 in File CelerTimelock.sol

```
NaN /*@CTK "transfer success case"
NaN   @post __return == this.balances[_owner]
```

```
NaN */
```

Line 352-354 in File CelerTimelock.sol

```
NaN function balanceOf(address _owner) public view returns (uint256) {
NaN   return balances[_owner];
NaN }
```

The code meets the specification

Detail for Request 19: pause

08, Sep 2018

Posted by CTK report generator

4.2ms

Line 480-483 in File CelerTimelock.sol

```
NaN /*@CTK "pause"
NaN   @tag assume_completion
NaN   @post post(this).paused == true
NaN */
```

Line 484-487 in File CelerTimelock.sol

```
NaN function pause() onlyOwner whenNotPaused public {
NaN   paused = true;
NaN   emit Pause();
NaN }
```

The code meets the specification

Detail for Request 20: onlyOwner_pause

08, Sep 2018

Posted by CTK report generator

2.6ms

Line 389-391 in File CelerTimelock.sol

```
NaN /*@CTK "onlyOwner"
NaN   @post msg.sender != owner -> __reverted
NaN */
```

Line 484-487 in File CelerTimelock.sol

```
NaN function pause() onlyOwner whenNotPaused public {
```

```
NaN   paused = true;
NaN   emit Pause();
NaN }
```

The code meets the specification

Detail for Request 21: whenNotPaused_pause

08, Sep 2018

Posted by CTK report generator

0.8ms

Line 458-460 in File CelerTimelock.sol

```
NaN /*@CTK "whenNotPaused"
NaN   @post this.paused -> __reverted
NaN */
```

Line 484-487 in File CelerTimelock.sol

```
NaN function pause() onlyOwner whenNotPaused public {
NaN   paused = true;
NaN   emit Pause();
NaN }
```

The code meets the specification

Detail for Request 22: unpause

08, Sep 2018

Posted by CTK report generator

2.1ms

Line 492-495 in File CelerTimelock.sol

```
NaN /*@CTK "unpause"
NaN   @tag assume_completion
NaN   @post post(this).paused == false
NaN */
```

Line 496-499 in File CelerTimelock.sol

```
NaN function unpause() onlyOwner whenPaused public {
NaN   paused = false;
NaN   emit Unpause();
NaN }
```

The code meets the specification

Detail for Request 23: onlyOwner_unpause

08, Sep 2018

1.7ms

Posted by CTK report generator

Line 389-391 in File CelerTimelock.sol

```
NaN /*@CTK "onlyOwner"
NaN   @post msg.sender != owner -> __reverted
NaN */
```

Line 496-499 in File CelerTimelock.sol

```
NaN function unpause() onlyOwner whenPaused public {
NaN   paused = false;
NaN   emit Unpause();
NaN }
```

The code meets the specification

Detail for Request 24: whenPaused_unpause

08, Sep 2018

0.8ms

Posted by CTK report generator

Line 469-471 in File CelerTimelock.sol

```
NaN /*@CTK "whenPaused"
NaN   @post !this.paused -> __reverted
NaN */
```

Line 496-499 in File CelerTimelock.sol

```
NaN function unpause() onlyOwner whenPaused public {
NaN   paused = false;
NaN   emit Unpause();
NaN }
```

The code meets the specification

Detail for Request 25: transferFrom success

08, Sep 2018

Posted by CTK report generator

204ms

Line 521-530 in File CelerTimelock.sol

```
NaN /*@CTK "transferFrom success"
NaN   @tag assume_completion
NaN   @pre _to != address(0)
NaN   @pre _value <= balances[_from]
NaN   @pre _value <= allowed[_from][msg.sender]
NaN   @post _from != _to -> __post.balances[_from] == balances[_from] - _value
NaN   @post _from != _to -> __post.balances[_to] == balances[_to] + _value
NaN   @post __post.allowed[_from][msg.sender] == allowed[_from][msg.sender] - _value
NaN   @post __return == true
NaN */
```

Line 543-560 in File CelerTimelock.sol

```
NaN function transferFrom(
NaN   address _from,
NaN   address _to,
NaN   uint256 _value
NaN )
NaN   public
NaN   returns (bool)
NaN {
NaN   require(_to != address(0));
NaN   require(_value <= balances[_from]);
NaN   require(_value <= allowed[_from][msg.sender]);
NaN
NaN   balances[_from] = balances[_from].sub(_value);
NaN   balances[_to] = balances[_to].add(_value);
NaN   allowed[_from][msg.sender] = allowed[_from][msg.sender].sub(_value);
NaN   emit Transfer(_from, _to, _value);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 26: transferFrom failure case 1: no enough balance

08, Sep 2018

Posted by CTK report generator

22ms

Line 531-534 in File CelerTimelock.sol

```
NaN /*@CTK "transferFrom failure case 1: no enough balance"
NaN   @pre balances[_from] < _value
NaN   @post __reverted
NaN */
```

Line 543-560 in File CelerTimelock.sol

```
NaN function transferFrom(
NaN   address _from,
NaN   address _to,
NaN   uint256 _value
NaN )
NaN   public
NaN   returns (bool)
NaN {
NaN   require(_to != address(0));
NaN   require(_value <= balances[_from]);
NaN   require(_value <= allowed[_from][msg.sender]);
NaN
NaN   balances[_from] = balances[_from].sub(_value);
NaN   balances[_to] = balances[_to].add(_value);
NaN   allowed[_from][msg.sender] = allowed[_from][msg.sender].sub(_value);
NaN   emit Transfer(_from, _to, _value);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 27: transferFrom failure case 2: no enough allowance

08, Sep 2018

Posted by CTK report generator

19.5ms

Line 535-538 in File CelerTimelock.sol

```
NaN /*@CTK "transferFrom failure case 2: no enough allowance"
NaN   @pre allowed[_from][msg.sender] < _value
NaN   @post __reverted
NaN */
```

Line 543-560 in File CelerTimelock.sol

```
NaN function transferFrom(
NaN   address _from,
NaN   address _to,
NaN   uint256 _value
NaN )
NaN   public
NaN   returns (bool)
NaN {
NaN   require(_to != address(0));
NaN   require(_value <= balances[_from]);
NaN   require(_value <= allowed[_from][msg.sender]);
NaN
NaN   balances[_from] = balances[_from].sub(_value);
NaN   balances[_to] = balances[_to].add(_value);
NaN   allowed[_from][msg.sender] = allowed[_from][msg.sender].sub(_value);
NaN   emit Transfer(_from, _to, _value);
```

```
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 28: transferFrom failure case 3: _to is 0

08, Sep 2018

Posted by CTK report generator

0.9ms

Line 539-542 in File CelerTimelock.sol

```
NaN /*@CTK "transferFrom failure case 3: _to is 0"
NaN   @pre _to == address(0)
NaN   @post __reverted
NaN */
```

Line 543-560 in File CelerTimelock.sol

```
NaN function transferFrom(
NaN   address _from,
NaN   address _to,
NaN   uint256 _value
NaN )
NaN   public
NaN   returns (bool)
NaN {
NaN   require(_to != address(0));
NaN   require(_value <= balances[_from]);
NaN   require(_value <= allowed[_from][msg.sender]);
NaN
NaN   balances[_from] = balances[_from].sub(_value);
NaN   balances[_to] = balances[_to].add(_value);
NaN   allowed[_from][msg.sender] = allowed[_from][msg.sender].sub(_value);
NaN   emit Transfer(_from, _to, _value);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 29: If method completes, integer overflow would not happen.

08, Sep 2018

Posted by CTK report generator

0.5ms

Line 572 in File CelerTimelock.sol

```
NaN //@CTK NO_OVERFLOW
```

Line 578-582 in File CelerTimelock.sol

```
NaN function approve(address _spender, uint256 _value) public returns (bool) {
NaN   allowed[msg.sender][_spender] = _value;
NaN   emit Approval(msg.sender, _spender, _value);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 30: Method will not encounter an assertion failure.

08, Sep 2018

Posted by CTK report generator

0.7ms

Line 573 in File CelerTimelock.sol

```
NaN //@CTK NO_ASF
```

Line 578-582 in File CelerTimelock.sol

```
NaN function approve(address _spender, uint256 _value) public returns (bool) {
NaN   allowed[msg.sender][_spender] = _value;
NaN   emit Approval(msg.sender, _spender, _value);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 31: approve transfer allowance

08, Sep 2018

Posted by CTK report generator

2.5ms

Line 574-577 in File CelerTimelock.sol

```
NaN /*@CTK "approve transfer allowance"
NaN   @post post(this).allowed[msg.sender][_spender] == _value
```

```
NaN   @post __return == true
NaN */
```

Line 578-582 in File CelerTimelock.sol

```
NaN function approve(address _spender, uint256 _value) public returns (bool) {
NaN   allowed[msg.sender][_spender] = _value;
NaN   emit Approval(msg.sender, _spender, _value);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 32: get the allowance

08, Sep 2018

Posted by CTK report generator

0.5ms

Line 590-593 in File CelerTimelock.sol

```
NaN /*@CTK "get the allowance"
NaN   @post __return == allowed[_owner][_spender]
NaN   @post this == post(this)
NaN */
```

Line 594-603 in File CelerTimelock.sol

```
NaN function allowance(
NaN   address _owner,
NaN   address _spender
NaN  )
NaN   public
NaN   view
NaN   returns (uint256)
NaN {
NaN   return allowed[_owner][_spender];
NaN }
```

The code meets the specification

Detail for Request 33: increaseApproval ok

08, Sep 2018

Posted by CTK report generator

1.9ms

Line 614-618 in File CelerTimelock.sol

```
NaN /*@CTK "increaseApproval ok"
NaN   @tag assume_completion
NaN   @post post(this).allowed[msg.sender][_spender] == this.allowed[msg.sender][_spenc
NaN   @post __return == true
NaN */
```

Line 619-630 in File CelerTimelock.sol

```
NaN function increaseApproval(
NaN   address _spender,
NaN   uint256 _addedValue
NaN )
NaN   public
NaN   returns (bool)
NaN {
NaN   allowed[msg.sender][_spender] = (
NaN     allowed[msg.sender][_spender].add(_addedValue));
NaN   emit Approval(msg.sender, _spender, allowed[msg.sender][_spender]);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 34: decreaseApproval case if

08, Sep 2018

Posted by CTK report generator

2.1ms

Line 641-646 in File CelerTimelock.sol

```
NaN /*@CTK "decreaseApproval case if"
NaN   @tag assume_completion
NaN   @pre _subtractedValue > allowed[msg.sender][_spender]
NaN   @post post(this).allowed[msg.sender][_spender] == 0
NaN   @post __return == true
NaN */
```

Line 653-668 in File CelerTimelock.sol

```
NaN function decreaseApproval(
NaN   address _spender,
NaN   uint256 _subtractedValue
NaN )
NaN   public
NaN   returns (bool)
NaN {
NaN   uint256 oldValue = allowed[msg.sender][_spender];
NaN   if (_subtractedValue > oldValue) {
NaN     allowed[msg.sender][_spender] = 0;
NaN   } else {
NaN     allowed[msg.sender][_spender] = oldValue.sub(_subtractedValue);
NaN   }
NaN   emit Approval(msg.sender, _spender, allowed[msg.sender][_spender]);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 35: decreaseApproval case else

08, Sep 2018

Posted by CTK report generator

3.1ms

Line 647-652 in File CelerTimelock.sol

```
NaN /*@CTK "decreaseApproval case else"
NaN   @tag assume_completion
NaN   @pre _subtractedValue <= allowed[msg.sender][_spender]
NaN   @post post(this).allowed[msg.sender][_spender] == this.allowed[msg.sender][_spenc
NaN   @post __return == true
NaN */
```

Line 653-668 in File CelerTimelock.sol

```
NaN function decreaseApproval(
NaN   address _spender,
NaN   uint256 _subtractedValue
NaN )
NaN   public
NaN   returns (bool)
NaN {
NaN   uint256 oldValue = allowed[msg.sender][_spender];
NaN   if (_subtractedValue > oldValue) {
NaN     allowed[msg.sender][_spender] = 0;
NaN   } else {
NaN     allowed[msg.sender][_spender] = oldValue.sub(_subtractedValue);
NaN   }
NaN   emit Approval(msg.sender, _spender, allowed[msg.sender][_spender]);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 36: addAddressToWhitelist

08, Sep 2018

Posted by CTK report generator

1.3ms

Line 703-707 in File CelerTimelock.sol

```
NaN /*@CTK "addAddressToWhitelist"
NaN   @tag assume_completion
NaN   @post post(this).whitelisted[_operator] == true
NaN   @post __return == true
NaN */
```

Line 708-716 in File CelerTimelock.sol

```
NaN function addAddressToWhitelist(address _operator)
NaN   onlyOwner
NaN   public
NaN   returns (bool)
NaN {
NaN   whitelisted[_operator] = true;
NaN   emit WhitelistAdded(_operator);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 37: onlyOwner_addAddressToWhitelist

08, Sep 2018

Posted by CTK report generator

0.7ms

Line 389-391 in File CelerTimelock.sol

```
NaN /*@CTK "onlyOwner"
NaN   @post msg.sender != owner -> __reverted
NaN */
```

Line 708-716 in File CelerTimelock.sol

```
NaN function addAddressToWhitelist(address _operator)
NaN   onlyOwner
NaN   public
NaN   returns (bool)
NaN {
NaN   whitelisted[_operator] = true;
NaN   emit WhitelistAdded(_operator);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 38: whitelist happy case

08, Sep 2018

Posted by CTK report generator

0.3ms

Line 721-724 in File CelerTimelock.sol

```
NaN /*@CTK "whitelist happy case"
NaN   @post __return == this.whitelisted[_operator]
NaN   @post post(this) == this
NaN */
```

Line 725-732 in File CelerTimelock.sol

```
NaN function whitelist(address _operator)
NaN   public
NaN   view
NaN   returns (bool)
NaN {
NaN   bool result = whitelisted[_operator];
NaN   return result;
NaN }
```

The code meets the specification

Detail for Request 39: addAddressesToWhitelist happy case

08, Sep 2018

Posted by CTK report generator

5.9ms

Line 740-744 in File CelerTimelock.sol

```
NaN /*@CTK "addAddressesToWhitelist happy case"
NaN   @tag assume_completion
NaN   @post forall i: uint. (i >= 0 /\ i < _operators.length) -> post(this).whitelistec
NaN   @post __return == true
NaN */
```

Line 745-761 in File CelerTimelock.sol

```
NaN function addAddressesToWhitelist(address[] _operators)
NaN   onlyOwner
NaN   public
NaN   returns (bool)
NaN {
NaN   /*@CTK addAddressesToWhitelist_forloop
NaN     @inv i <= _operators.length
NaN     @inv _operators == _operators__pre
NaN     @inv forall j: uint. (j >= 0 /\ j < i) -> this.whitelisted[_operators[j]] == tr
NaN     @post i == _operators.length
NaN     @post !__should_return
NaN   */
NaN   for (uint256 i = 0; i < _operators.length; i++) {
NaN     require(addAddressToWhitelist(_operators[i]));
NaN   }
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 40: onlyOwner_addAddressesToWhitelist

08, Sep 2018

Posted by CTK report generator

1ms

Line 389-391 in File CelerTimelock.sol

```
NaN /*@CTK "onlyOwner"
NaN   @post msg.sender != owner -> __reverted
NaN */
```

Line 745-761 in File CelerTimelock.sol

```
NaN function addAddressesToWhitelist(address[] _operators)
NaN   onlyOwner
NaN   public
NaN   returns (bool)
NaN {
NaN   /*@CTK addAddressesToWhitelist_forloop
NaN     @inv i <= _operators.length
NaN     @inv _operators == _operators__pre
NaN     @inv forall j: uint. (j >= 0 /\ j < i) -> this.whitelisted[_operators[j]] == tr
NaN     @post i == _operators.length
NaN     @post !__should_return
NaN   */
NaN   for (uint256 i = 0; i < _operators.length; i++) {
NaN     require(addAddressToWhitelist(_operators[i]));
NaN   }
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 41: removeAddressFromWhitelist happy case

08, Sep 2018

Posted by CTK report generator

1.3ms

Line 769-773 in File CelerTimelock.sol

```
NaN /*@CTK "removeAddressFromWhitelist happy case"
NaN   @tag assume_completion
NaN   @post post(this).whitelisted[_operator] == false
```

```
NaN   @post __return == true
NaN */
```

Line 774-782 in File CelerTimelock.sol

```
NaN function removeAddressFromWhitelist(address _operator)
NaN   onlyOwner
NaN   public
NaN   returns (bool)
NaN {
NaN   whitelisted[_operator] = false;
NaN   emit WhitelistRemoved(_operator);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 42: onlyOwner_removeAddressFromWhitelist

08, Sep 2018

Posted by CTK report generator

0.7ms

Line 389-391 in File CelerTimelock.sol

```
NaN /*@CTK "onlyOwner"
NaN   @post msg.sender != owner -> __reverted
NaN */
```

Line 774-782 in File CelerTimelock.sol

```
NaN function removeAddressFromWhitelist(address _operator)
NaN   onlyOwner
NaN   public
NaN   returns (bool)
NaN {
NaN   whitelisted[_operator] = false;
NaN   emit WhitelistRemoved(_operator);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 43: removeAddressesFromWhitelist happy case

08, Sep 2018

Posted by CTK report generator

4.5ms

Line 790-794 in File CelerTimelock.sol

```
NaN /*@CTK "removeAddressesFromWhitelist happy case"
NaN   @tag assume_completion
NaN   @post forall i: uint. (i >= 0 /\ i < _operators.length) -> post(this).whitelistec
NaN   @post __return == true
NaN */
```

Line 795-811 in File CelerTimelock.sol

```
NaN function removeAddressesFromWhitelist(address[] _operators)
NaN   onlyOwner
NaN   public
NaN   returns (bool)
NaN {
NaN   /*@CTK removeAddressesFromWhitelist_forLoop
NaN     @inv i <= _operators.length
NaN     @inv _operators == _operators__pre
NaN     @inv forall j: uint. (j >= 0 /\ j < i) -> this.whitelisted[_operators[j]] == fa
NaN     @post i == _operators.length
NaN     @post !__should_return
NaN   */
NaN   for (uint256 i = 0; i < _operators.length; i++) {
NaN     require(removeAddressFromWhitelist(_operators[i]));
NaN   }
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 44: onlyOwner_removeAddressesFromWhitelist

08, Sep 2018

Posted by CTK report generator

1.2ms

Line 389-391 in File CelerTimelock.sol

```
NaN /*@CTK "onlyOwner"
NaN   @post msg.sender != owner -> __reverted
NaN */
```

Line 795-811 in File CelerTimelock.sol

```
NaN function removeAddressesFromWhitelist(address[] _operators)
NaN   onlyOwner
NaN   public
NaN   returns (bool)
NaN {
NaN   /*@CTK removeAddressesFromWhitelist_forLoop
NaN     @inv i <= _operators.length
NaN     @inv _operators == _operators__pre
NaN     @inv forall j: uint. (j >= 0 /\ j < i) -> this.whitelisted[_operators[j]] == fa
NaN     @post i == _operators.length
NaN     @post !__should_return
NaN   */
```

```
NaN   for (uint256 i = 0; i < _operators.length; i++) {
NaN     require(removeAddressFromWhitelist(_operators[i]));
NaN   }
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 45: PausableToken transfer success case

08, Sep 2018

Posted by CTK report generator

160.6ms

Line 822-827 in File CelerTimelock.sol

```
NaN /*@CTK "PausableToken transfer success case"
NaN   @tag assume_completion
NaN   @post _to != msg.sender -> __post.balances[msg.sender] == balances[msg.sender] -
NaN   @post _to != msg.sender -> __post.balances[_to] == balances[_to] + _value
NaN   @post __return == true
NaN */
```

Line 828-837 in File CelerTimelock.sol

```
NaN function transfer(
NaN   address _to,
NaN   uint256 _value
NaN )
NaN   public
NaN   whenNotPaused
NaN   returns (bool)
NaN {
NaN   return super.transfer(_to, _value);
NaN }
```

The code meets the specification

Detail for Request 46: whenNotPaused_transfer

08, Sep 2018

Posted by CTK report generator

2.1ms

Line 458-460 in File CelerTimelock.sol

```
NaN /*@CTK "whenNotPaused"
NaN   @post this.paused -> __reverted
NaN */
```

Line 828-837 in File CelerTimelock.sol

```
NaN function transfer(
NaN   address _to,
NaN   uint256 _value
NaN )
NaN   public
NaN   whenNotPaused
NaN   returns (bool)
NaN {
NaN   return super.transfer(_to, _value);
NaN }
```

The code meets the specification

Detail for Request 47: PausableToken transferFrom success

08, Sep 2018

Posted by CTK report generator

285.7ms

Line 839-845 in File CelerTimelock.sol

```
NaN /*@CTK "PausableToken transferFrom success"
NaN   @tag assume_completion
NaN   @post _from != _to -> __post.balances[_from] == balances[_from] - _value
NaN   @post _from != _to -> __post.balances[_to] == balances[_to] + _value
NaN   @post __post.allowed[_from][msg.sender] == allowed[_from][msg.sender] - _value
NaN   @post __return == true
NaN */
```

Line 846-856 in File CelerTimelock.sol

```
NaN function transferFrom(
NaN   address _from,
NaN   address _to,
NaN   uint256 _value
NaN )
NaN   public
NaN   whenNotPaused
NaN   returns (bool)
NaN {
NaN   return super.transferFrom(_from, _to, _value);
NaN }
```

The code meets the specification

Detail for Request 48: whenNotPaused_transferFrom

08, Sep 2018

Posted by CTK report generator

1.6ms

Line 458-460 in File CelerTimelock.sol

```
NaN /*@CTK "whenNotPaused"
NaN   @post this.paused -> __reverted
NaN */
```

Line 846-856 in File CelerTimelock.sol

```
NaN function transferFrom(
NaN   address _from,
NaN   address _to,
NaN   uint256 _value
NaN )
NaN   public
NaN   whenNotPaused
NaN   returns (bool)
NaN {
NaN   return super.transferFrom(_from, _to, _value);
NaN }
```

The code meets the specification

Detail for Request 49: PausableToken approve transfer allowance

08, Sep 2018

Posted by CTK report generator

3.4ms

Line 858-862 in File CelerTimelock.sol

```
NaN /*@CTK "PausableToken approve transfer allowance"
NaN   @pre paused == false
NaN   @post post(this).allowed[msg.sender][_spender] == _value
NaN   @post __return == true
NaN */
```

Line 863-872 in File CelerTimelock.sol

```
NaN function approve(
NaN   address _spender,
NaN   uint256 _value
NaN )
NaN   public
NaN   whenNotPaused
NaN   returns (bool)
NaN {
NaN   return super.approve(_spender, _value);
NaN }
```

The code meets the specification

Detail for Request 50: whenNotPaused_approve

08, Sep 2018

Posted by CTK report generator

1.1ms

Line 458-460 in File CelerTimelock.sol

```
NaN /*@CTK "whenNotPaused"
NaN   @post this.paused -> __reverted
NaN */
```

Line 863-872 in File CelerTimelock.sol

```
NaN function approve(
NaN   address _spender,
NaN   uint256 _value
NaN )
NaN   public
NaN   whenNotPaused
NaN   returns (bool)
NaN {
NaN   return super.approve(_spender, _value);
NaN }
```

The code meets the specification

Detail for Request 51: increaseApproval ok

08, Sep 2018

Posted by CTK report generator

3.7ms

Line 873-876 in File CelerTimelock.sol

```
NaN /*@CTK "increaseApproval ok"
NaN   @tag assume_completion
NaN   @post post(this).allowed[msg.sender][_spender] == this.allowed[msg.sender][_spenc
NaN */
```

Line 877-886 in File CelerTimelock.sol

```
NaN function increaseApproval(
NaN   address _spender,
NaN   uint _addedValue
```

```
NaN )
NaN   public
NaN   whenNotPaused
NaN   returns (bool success)
NaN {
NaN   return super.increaseApproval(_spender, _addedValue);
NaN }
```

The code meets the specification

Detail for Request 52: whenNotPaused_increaseApproval

08, Sep 2018

Posted by CTK report generator

1.2ms

Line 458-460 in File CelerTimelock.sol

```
NaN /*@CTK "whenNotPaused"
NaN   @post this.paused -> __reverted
NaN */
```

Line 877-886 in File CelerTimelock.sol

```
NaN function increaseApproval(
NaN   address _spender,
NaN   uint _addedValue
NaN )
NaN   public
NaN   whenNotPaused
NaN   returns (bool success)
NaN {
NaN   return super.increaseApproval(_spender, _addedValue);
NaN }
```

The code meets the specification

Detail for Request 53: decreaseApproval case if

08, Sep 2018

Posted by CTK report generator

25.8ms

Line 888-892 in File CelerTimelock.sol

```
NaN /*@CTK "decreaseApproval case if"
NaN   @tag assume_completion
```

```
NaN   @pre _subtractedValue > allowed[msg.sender][_spender]
NaN   @post post(this).allowed[msg.sender][_spender] == 0
NaN */
```

Line 898-907 in File CelerTimelock.sol

```
NaN function decreaseApproval(
NaN   address _spender,
NaN   uint _subtractedValue
NaN )
NaN   public
NaN   whenNotPaused
NaN   returns (bool success)
NaN {
NaN   return super.decreaseApproval(_spender, _subtractedValue);
NaN }
```

The code meets the specification

Detail for Request 54: decreaseApproval case else

08, Sep 2018

Posted by CTK report generator

47.1ms

Line 893-897 in File CelerTimelock.sol

```
NaN /*@CTK "decreaseApproval case else"
NaN   @tag assume_completion
NaN   @pre _subtractedValue <= allowed[msg.sender][_spender]
NaN   @post post(this).allowed[msg.sender][_spender] == this.allowed[msg.sender][_spenc
NaN */
```

Line 898-907 in File CelerTimelock.sol

```
NaN function decreaseApproval(
NaN   address _spender,
NaN   uint _subtractedValue
NaN )
NaN   public
NaN   whenNotPaused
NaN   returns (bool success)
NaN {
NaN   return super.decreaseApproval(_spender, _subtractedValue);
NaN }
```

The code meets the specification

Detail for Request 55: whenNotPaused_decreaseApproval

08, Sep 2018

Posted by CTK report generator

1.4ms

Line 458-460 in File CelerTimelock.sol

```
NaN /*@CTK "whenNotPaused"
NaN   @post this.paused -> __reverted
NaN */
```

Line 898-907 in File CelerTimelock.sol

```
NaN function decreaseApproval(
NaN   address _spender,
NaN   uint _subtractedValue
NaN )
NaN   public
NaN   whenNotPaused
NaN   returns (bool success)
NaN {
NaN   return super.decreaseApproval(_spender, _subtractedValue);
NaN }
```

The code meets the specification

Detail for Request 56: CelerToken constructor

08, Sep 2018

Posted by CTK report generator

2.7ms

Line 944-947 in File CelerTimelock.sol

```
NaN /*@CTK "CelerToken constructor"
NaN   @post post(this).totalSupply_ == INITIAL_SUPPLY
NaN   @post post(this).balances[msg.sender] == INITIAL_SUPPLY
NaN */
```

Line 948-951 in File CelerTimelock.sol

```
NaN constructor() public {
NaN   totalSupply_ = INITIAL_SUPPLY;
NaN   balances[msg.sender] = INITIAL_SUPPLY;
NaN }
```

The code meets the specification

Detail for Request 57: CelerToken transfer success case

08, Sep 2018

Posted by CTK report generator

223.2ms

Line 957-962 in File CelerTimelock.sol

```
NaN /*@CTK "CelerToken transfer success case"
NaN   @tag assume_completion
NaN   @post _to != msg.sender -> __post.balances[msg.sender] == balances[msg.sender] -
NaN   @post _to != msg.sender -> __post.balances[_to] == balances[_to] + _value
NaN   @post __return == true
NaN */
```

Line 963-973 in File CelerTimelock.sol

```
NaN function transfer(
NaN   address _to,
NaN   uint256 _value
NaN )
NaN   public
NaN   onlyIfTransferable
NaN   onlyValidReceiver(_to)
NaN   returns (bool)
NaN {
NaN   return super.transfer(_to, _value);
NaN }
```

The code meets the specification

Detail for Request 58: onlyValidReceiver_transfer

08, Sep 2018

Posted by CTK report generator

7ms

Line 926-929 in File CelerTimelock.sol

```
NaN /*@CTK onlyValidReceiver
NaN   @pre this.transferOpened == false /\ whitelisted[msg.sender] == false /\ msg.senc
NaN   @post __reverted
NaN */
```

Line 963-973 in File CelerTimelock.sol

```
NaN function transfer(
NaN   address _to,
NaN   uint256 _value
NaN )
NaN   public
NaN   onlyIfTransferable
NaN   onlyValidReceiver(_to)
NaN   returns (bool)
```

```
NaN {
NaN   return super.transfer(_to, _value);
NaN }
```

The code meets the specification

Detail for Request 59: onlyValidReceiver_transfer

08, Sep 2018

Posted by CTK report generator

7ms

Line 935-937 in File CelerTimelock.sol

```
NaN /*@CTK onlyValidReceiver
NaN   @post _to == address(this) -> __reverted
NaN */
```

Line 963-973 in File CelerTimelock.sol

```
NaN function transfer(
NaN   address _to,
NaN   uint256 _value
NaN )
NaN   public
NaN   onlyIfTransferable
NaN   onlyValidReceiver(_to)
NaN   returns (bool)
NaN {
NaN   return super.transfer(_to, _value);
NaN }
```

The code meets the specification

Detail for Request 60: transferFrom transferFrom success

08, Sep 2018

Posted by CTK report generator

319.5ms

Line 979-985 in File CelerTimelock.sol

```
NaN /*@CTK "transferFrom transferFrom success"
NaN   @tag assume_completion
NaN   @post _from != _to -> __post.balances[_from] == balances[_from] - _value
NaN   @post _from != _to -> __post.balances[_to] == balances[_to] + _value
NaN   @post __post.allowed[_from][msg.sender] == allowed[_from][msg.sender] - _value
```

```
NaN   @post __return == true
NaN */
```

Line 986-997 in File CelerTimelock.sol

```
NaN function transferFrom(
NaN   address _from,
NaN   address _to,
NaN   uint256 _value
NaN )
NaN   public
NaN   onlyIfTransferable
NaN   onlyValidReceiver(_to)
NaN   returns (bool)
NaN {
NaN   return super.transferFrom(_from, _to, _value);
NaN }
```

The code meets the specification

Detail for Request 61: onlyValidReceiver_transferFrom

08, Sep 2018

Posted by CTK report generator

9.7ms

Line 926-929 in File CelerTimelock.sol

```
NaN /*@CTK onlyValidReceiver
NaN   @pre this.transferOpened == false /\ whitelisted[msg.sender] == false /\ msg.senc
NaN   @post __reverted
NaN */
```

Line 986-997 in File CelerTimelock.sol

```
NaN function transferFrom(
NaN   address _from,
NaN   address _to,
NaN   uint256 _value
NaN )
NaN   public
NaN   onlyIfTransferable
NaN   onlyValidReceiver(_to)
NaN   returns (bool)
NaN {
NaN   return super.transferFrom(_from, _to, _value);
NaN }
```

The code meets the specification

Detail for Request 62: onlyValidReceiver_transferFrom

08, Sep 2018

9.7ms

Posted by CTK report generator

Line 935-937 in File CelerTimelock.sol

```
NaN /*@CTK onlyValidReceiver
NaN   @post _to == address(this) -> __reverted
NaN */
```

Line 986-997 in File CelerTimelock.sol

```
NaN function transferFrom(
NaN   address _from,
NaN   address _to,
NaN   uint256 _value
NaN )
NaN   public
NaN   onlyIfTransferable
NaN   onlyValidReceiver(_to)
NaN   returns (bool)
NaN {
NaN   return super.transferFrom(_from, _to, _value);
NaN }
```

The code meets the specification

Detail for Request 63: openTransfer ok

08, Sep 2018

2.6ms

Posted by CTK report generator

Line 1003-1006 in File CelerTimelock.sol

```
NaN /*@CTK "openTransfer ok"
NaN   @tag assume_completion
NaN   @post post(this).transferOpened == true
NaN */
```

Line 1007-1009 in File CelerTimelock.sol

```
NaN function openTransfer() external onlyOwner {
NaN   transferOpened = true;
NaN }
```

The code meets the specification

Detail for Request 64: onlyOwner_openTransfer

08, Sep 2018

Posted by CTK report generator

1.6ms

Line 389-391 in File CelerTimelock.sol

```
NaN /*@CTK "onlyOwner"
NaN   @post msg.sender != owner -> __reverted
NaN */
```

Line 1007-1009 in File CelerTimelock.sol

```
NaN function openTransfer() external onlyOwner {
NaN   transferOpened = true;
NaN }
```

The code meets the specification

Detail for Request 65: activated violation from whenNotActivated

08, Sep 2018

Posted by CTK report generator

1.2ms

Line 1149-1152 in File CelerTimelock.sol

```
NaN /*@CTK "activated violation from whenNotActivated"
NaN   @pre isActivated == true
NaN   @post __reverted
NaN */
```

Line 1153-1158 in File CelerTimelock.sol

```
NaN function activateNow() onlyOwner whenNotActivated public returns (bool) {
NaN   isActivated = true;
NaN   startTime = now;
NaN   emit Activate(startTime);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 66: onlyOwner_activateNow

08, Sep 2018

Posted by CTK report generator

3.3ms

Line 389-391 in File CelerTimelock.sol

```
NaN /*@CTK "onlyOwner"
NaN   @post msg.sender != owner -> __reverted
NaN */
```

Line 1153-1158 in File CelerTimelock.sol

```
NaN function activateNow() onlyOwner whenNotActivated public returns (bool) {
NaN   isActivated = true;
NaN   startTime = now;
NaN   emit Activate(startTime);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 67: activateWithTime

08, Sep 2018

Posted by CTK report generator

21ms

Line 1160-1164 in File CelerTimelock.sol

```
NaN /*@CTK "activateWithTime"
NaN   @tag assume_completion
NaN   @post __post.isActivated == true
NaN   @post __return == true
NaN */
```

Line 1173-1180 in File CelerTimelock.sol

```
NaN function activateWithTime(uint256 _startTime) onlyOwner whenNotActivated public ret
NaN   require(_startTime > 0);
NaN
NaN   isActivated = true;
NaN   startTime = _startTime;
NaN   emit Activate(startTime);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 68: start time smaller equal to 0

08, Sep 2018

Posted by CTK report generator

15.8ms

Line 1165-1168 in File CelerTimelock.sol

```
NaN /*@CTK "start time smaller equal to 0"
NaN   @pre _startTime <= 0
NaN   @post __reverted
NaN */
```

Line 1173-1180 in File CelerTimelock.sol

```
NaN function activateWithTime(uint256 _startTime) onlyOwner whenNotActivated public ret
NaN   require(_startTime > 0);
NaN
NaN   isActivated = true;
NaN   startTime = _startTime;
NaN   emit Activate(startTime);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 69: activated violation from whenNotActivated_2

08, Sep 2018

Posted by CTK report generator

1.2ms

Line 1169-1172 in File CelerTimelock.sol

```
NaN /*@CTK "activated violation from whenNotActivated_2"
NaN   @pre isActivated == true
NaN   @post __reverted
NaN */
```

Line 1173-1180 in File CelerTimelock.sol

```
NaN function activateWithTime(uint256 _startTime) onlyOwner whenNotActivated public ret
NaN   require(_startTime > 0);
NaN
NaN   isActivated = true;
NaN   startTime = _startTime;
NaN   emit Activate(startTime);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 70: onlyOwner_activateWithTime

08, Sep 2018

Posted by CTK report generator

4ms

Line 389-391 in File CelerTimelock.sol

```
NaN  /*@CTK "onlyOwner"
NaN    @post msg.sender != owner -> __reverted
NaN  */
```

Line 1173-1180 in File CelerTimelock.sol

```
NaN  function activateWithTime(uint256 _startTime) onlyOwner whenNotActivated public ret
NaN    require(_startTime > 0);
NaN
NaN    isActivated = true;
NaN    startTime = _startTime;
NaN    emit Activate(startTime);
NaN    return true;
NaN  }
```

The code meets the specification

Detail for Request 71: beneficiary equal 0

08, Sep 2018

Posted by CTK report generator

4.1ms

Line 1182-1185 in File CelerTimelock.sol

```
NaN  /*@CTK "beneficiary equal 0"
NaN    @pre _newBeneficiary == address(0)
NaN    @post __reverted
NaN  */
```

Line 1195-1202 in File CelerTimelock.sol

```
NaN  function resetBeneficiary(address _newBeneficiary) onlyOwner public returns (bool)
NaN    require(_newBeneficiary != address(0));
NaN    require(_newBeneficiary != beneficiary);
NaN
NaN    beneficiary = _newBeneficiary;
NaN    emit ResetBeneficiary(beneficiary);
NaN    return true;
```

```
NaN }
```

The code meets the specification

Detail for Request 72: new beneficiary equal to origin

08, Sep 2018

Posted by CTK report generator

9.2ms

Line 1186-1189 in File CelerTimelock.sol

```
NaN /*@CTK "new beneficiary equal to origin"
NaN   @pre _newBeneficiary == beneficiary
NaN   @post __reverted
NaN */
```

Line 1195-1202 in File CelerTimelock.sol

```
NaN function resetBeneficiary(address _newBeneficiary) onlyOwner public returns (bool)
NaN   require(_newBeneficiary != address(0));
NaN   require(_newBeneficiary != beneficiary);
NaN
NaN   beneficiary = _newBeneficiary;
NaN   emit ResetBeneficiary(beneficiary);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 73: resetBeneficiary

08, Sep 2018

Posted by CTK report generator

7.5ms

Line 1190-1194 in File CelerTimelock.sol

```
NaN /*@CTK "resetBeneficiary"
NaN    @tag assume_completion
NaN    @post __return == true
NaN    @post __post.beneficiary == _newBeneficiary
NaN  */
```

Line 1195-1202 in File CelerTimelock.sol

```
NaN function resetBeneficiary(address _newBeneficiary) onlyOwner public returns (bool)
```

```
NaN   require(_newBeneficiary != address(0));
NaN   require(_newBeneficiary != beneficiary);
NaN
NaN   beneficiary = _newBeneficiary;
NaN   emit ResetBeneficiary(beneficiary);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 74: onlyOwner_resetBeneficiary

08, Sep 2018

Posted by CTK report generator

1ms

Line 389-391 in File CelerTimelock.sol

```
NaN /*@CTK "onlyOwner"
NaN   @post msg.sender != owner -> __reverted
NaN */
```

Line 1195-1202 in File CelerTimelock.sol

```
NaN function resetBeneficiary(address _newBeneficiary) onlyOwner public returns (bool)
NaN   require(_newBeneficiary != address(0));
NaN   require(_newBeneficiary != beneficiary);
NaN
NaN   beneficiary = _newBeneficiary;
NaN   emit ResetBeneficiary(beneficiary);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 75: getUnlockedStagesNumber not activated

08, Sep 2018

Posted by CTK report generator

5.2ms

Line 1204-1210 in File CelerTimelock.sol

```
NaN /*@CTK "getUnlockedStagesNumber not activated"
NaN   @tag assume_completion
NaN   @pre lockupTimes[2] > lockupTimes[1]
NaN   @pre lockupTimes[1] > lockupTimes[0]
```

```
NaN   @pre isActivated == false
NaN   @post __return == 0
NaN */
```

Line 1235-1251 in File CelerTimelock.sol

```
NaN function getUnlockedStagesNumber() public view returns (uint256) {
NaN   if (!isActivated) {
NaN     return 0;
NaN   }
NaN
NaN   if (now < (startTime + lockupTimes[0])) {
NaN     return 0;
NaN   } else if ((startTime + lockupTimes[0]) <= now && now < (startTime + lockupTimes[
NaN     return 1;
NaN   } else if ((startTime + lockupTimes[1]) <= now && now < (startTime + lockupTimes[
NaN     return 2;
NaN   } else if ((startTime + lockupTimes[2]) <= now) {
NaN     return 3;
NaN   } else {
NaN     assert(false);
NaN   }
NaN }
```

The code meets the specification

Detail for Request 76: getUnlockedStagesNumber 0 stage

08, Sep 2018

Posted by CTK report generator

3.3ms

Line 1211-1218 in File CelerTimelock.sol

```
NaN /*@CTK "getUnlockedStagesNumber 0 stage"
NaN   @tag assume_completion
NaN   @pre isActivated == true
NaN   @pre lockupTimes[2] > lockupTimes[1]
NaN   @pre lockupTimes[1] > lockupTimes[0]
NaN   @pre now < (startTime + lockupTimes[0])
NaN   @post __return == 0
NaN */
```

Line 1235-1251 in File CelerTimelock.sol

```
NaN function getUnlockedStagesNumber() public view returns (uint256) {
NaN   if (!isActivated) {
NaN     return 0;
NaN   }
NaN
NaN   if (now < (startTime + lockupTimes[0])) {
NaN     return 0;
NaN   } else if ((startTime + lockupTimes[0]) <= now && now < (startTime + lockupTimes[
NaN     return 1;
NaN   } else if ((startTime + lockupTimes[1]) <= now && now < (startTime + lockupTimes[
NaN     return 2;
NaN   } else if ((startTime + lockupTimes[2]) <= now) {
NaN     return 3;
NaN   } else {
```

```
NaN     assert(false);
NaN   }
NaN }
```

The code meets the specification

Detail for Request 77: getUnlockedStagesNumber 1st stage

08, Sep 2018

Posted by CTK report generator

5.4ms

Line 1219-1226 in File CelerTimelock.sol

```
NaN /*@CTK "getUnlockedStagesNumber 1st stage"
NaN   @tag assume_completion
NaN   @pre isActivated == true
NaN   @pre lockupTimes[2] > lockupTimes[1]
NaN   @pre lockupTimes[1] > lockupTimes[0]
NaN   @pre (startTime + lockupTimes[0]) <= now && now < (startTime + lockupTimes[1])
NaN   @post __return == 1
NaN */
```

Line 1235-1251 in File CelerTimelock.sol

```
NaN function getUnlockedStagesNumber() public view returns (uint256) {
NaN   if (!isActivated) {
NaN     return 0;
NaN   }
NaN
NaN   if (now < (startTime + lockupTimes[0])) {
NaN     return 0;
NaN   } else if ((startTime + lockupTimes[0]) <= now && now < (startTime + lockupTimes[
NaN     return 1;
NaN   } else if ((startTime + lockupTimes[1]) <= now && now < (startTime + lockupTimes[
NaN     return 2;
NaN   } else if ((startTime + lockupTimes[2]) <= now) {
NaN     return 3;
NaN   } else {
NaN     assert(false);
NaN   }
NaN }
```

The code meets the specification

Detail for Request 78: getUnlockedStagesNumber 3rd stage

08, Sep 2018
4.5ms

Posted by CTK report generator

Line 1227-1234 in File CelerTimelock.sol

```
NaN /*@CTK "getUnlockedStagesNumber 3rd stage"
NaN   @tag assume_completion
NaN   @pre isActivated == true
NaN   @pre (startTime + lockupTimes[0]) <= now
NaN   @pre (startTime + lockupTimes[1]) <= now
NaN   @pre (startTime + lockupTimes[2]) <= now
NaN   @post __return == 3
NaN */
```

Line 1235-1251 in File CelerTimelock.sol

```
NaN function getUnlockedStagesNumber() public view returns (uint256) {
NaN   if (!isActivated) {
NaN     return 0;
NaN   }
NaN
NaN   if (now < (startTime + lockupTimes[0])) {
NaN     return 0;
NaN   } else if ((startTime + lockupTimes[0]) <= now && now < (startTime + lockupTimes[
NaN     return 1;
NaN   } else if ((startTime + lockupTimes[1]) <= now && now < (startTime + lockupTimes[
NaN     return 2;
NaN   } else if ((startTime + lockupTimes[2]) <= now) {
NaN     return 3;
NaN   } else {
NaN     assert(false);
NaN   }
NaN }
```

The code meets the specification

CertiK

Certi Request Report

106 out of 106 specs are satisfied.

Detail for Request 0: Ownable constructor

08, Sep 2018

0.5ms

Posted by CTK report generator

Line 477-479 in File CelerCrowdsale.sol

```
NaN /*@CTK "Ownable constructor"
NaN   @post post(this).owner == msg.sender
NaN */
```

Line 480-482 in File CelerCrowdsale.sol

```
NaN constructor() public {
NaN   owner = msg.sender;
```

```
NaN }
```

The code meets the specification

Detail for Request 1: renounceOwnership

08, Sep 2018

1ms

Posted by CTK report generator

Line 501-504 in File CelerCrowdsale.sol

```
NaN /*@CTK "renounceOwnership"
NaN   @tag assume_completion
NaN   @post post(this).owner == address(0)
NaN */
```

Line 505-508 in File CelerCrowdsale.sol

```
NaN function renounceOwnership() public onlyOwner {
NaN   emit OwnershipRenounced(owner);
NaN   owner = address(0);
NaN }
```

The code meets the specification

Detail for Request 2: onlyOwner_renounceOwnership

08, Sep 2018

0.7ms

Posted by CTK report generator

Line 487-489 in File CelerCrowdsale.sol

```
NaN /*@CTK "onlyOwner"
NaN   @post msg.sender != owner -> __reverted
NaN */
```

Line 505-508 in File CelerCrowdsale.sol

```
NaN function renounceOwnership() public onlyOwner {
NaN   emit OwnershipRenounced(owner);
NaN   owner = address(0);
NaN }
```

The code meets the specification

Detail for Request 3: transferOwnership

08, Sep 2018

Posted by CTK report generator

3.5ms

Line 514-517 in File CelerCrowdsale.sol

```
NaN /*@CTK "transferOwnership"
NaN   @tag assume_completion
NaN   @post post(this).owner == _newOwner
NaN */
```

Line 518-520 in File CelerCrowdsale.sol

```
NaN function transferOwnership(address _newOwner) public onlyOwner {
NaN   _transferOwnership(_newOwner);
NaN }
```

The code meets the specification

Detail for Request 4: onlyOwner_transferOwnership

08, Sep 2018

Posted by CTK report generator

1.1ms

Line 487-489 in File CelerCrowdsale.sol

```
NaN /*@CTK "onlyOwner"
NaN   @post msg.sender != owner -> __reverted
NaN */
```

Line 518-520 in File CelerCrowdsale.sol

```
NaN function transferOwnership(address _newOwner) public onlyOwner {
NaN   _transferOwnership(_newOwner);
NaN }
```

The code meets the specification

Detail for Request 5: _transferOwnership success case

08, Sep 2018

Posted by CTK report generator

2.7ms

Line 526-529 in File CelerCrowdsale.sol

```
NaN /*@CTK "_transferOwnership success case"
NaN   @pre _newOwner != address(0)
NaN   @post post(this).owner == _newOwner
NaN */
```

Line 534-538 in File CelerCrowdsale.sol

```
NaN function _transferOwnership(address _newOwner) internal {
NaN   require(_newOwner != address(0));
NaN   emit OwnershipTransferred(owner, _newOwner);
NaN   owner = _newOwner;
NaN }
```

The code meets the specification

Detail for Request 6: _transferOwnership reverted case

08, Sep 2018

Posted by CTK report generator

0.6ms

Line 530-533 in File CelerCrowdsale.sol

```
NaN /*@CTK "_transferOwnership reverted case"
NaN   @pre _newOwner == address(0)
NaN   @post __reverted
NaN */
```

Line 534-538 in File CelerCrowdsale.sol

```
NaN function _transferOwnership(address _newOwner) internal {
NaN   require(_newOwner != address(0));
NaN   emit OwnershipTransferred(owner, _newOwner);
NaN   owner = _newOwner;
NaN }
```

The code meets the specification

Detail for Request 7: addAddressToWhitelist

08, Sep 2018

1.3ms

Posted by CTK report generator

Line 801-805 in File CelerCrowdsale.sol

```
NaN /*@CTK "addAddressToWhitelist"
NaN   @tag assume_completion
NaN   @post post(this).whitelisted[_operator] == true
NaN   @post __return == true
NaN */
```

Line 806-814 in File CelerCrowdsale.sol

```
NaN function addAddressToWhitelist(address _operator)
NaN   onlyOwner
NaN   public
NaN   returns (bool)
NaN {
NaN   whitelisted[_operator] = true;
NaN   emit WhitelistAdded(_operator);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 8: onlyOwner_addAddressToWhitelist

08, Sep 2018

0.7ms

Posted by CTK report generator

Line 487-489 in File CelerCrowdsale.sol

```
NaN /*@CTK "onlyOwner"
NaN   @post msg.sender != owner -> __reverted
NaN */
```

Line 806-814 in File CelerCrowdsale.sol

```
NaN function addAddressToWhitelist(address _operator)
NaN   onlyOwner
NaN   public
NaN   returns (bool)
NaN {
NaN   whitelisted[_operator] = true;
NaN   emit WhitelistAdded(_operator);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 9: whitelist happy case

08, Sep 2018

Posted by CTK report generator

0.3ms

Line 819-822 in File CelerCrowdsale.sol

```
NaN /*@CTK "whitelist happy case"
NaN   @post __return == this.whitelisted[_operator]
NaN   @post post(this) == this
NaN */
```

Line 823-830 in File CelerCrowdsale.sol

```
NaN function whitelist(address _operator)
NaN   public
NaN   view
NaN   returns (bool)
NaN {
NaN   bool result = whitelisted[_operator];
NaN   return result;
NaN }
```

The code meets the specification

Detail for Request 10: addAddressesToWhitelist happy case

08, Sep 2018

Posted by CTK report generator

5.9ms

Line 838-842 in File CelerCrowdsale.sol

```
NaN /*@CTK "addAddressesToWhitelist happy case"
NaN   @tag assume_completion
NaN   @post forall i: uint. (i >= 0 /\ i < _operators.length) -> post(this).whitelistec
NaN   @post __return == true
NaN */
```

Line 843-859 in File CelerCrowdsale.sol

```
NaN function addAddressesToWhitelist(address[] _operators)
NaN   onlyOwner
NaN   public
NaN   returns (bool)
NaN {
NaN   /*@CTK addAddressesToWhitelist_forloop
NaN     @inv i <= _operators.length
```

```
NaN     @inv _operators == _operators__pre
NaN     @inv forall j: uint. (j >= 0 /\ j < i) -> this.whitelisted[_operators[j]] == tr
NaN     @post i == _operators.length
NaN     @post !__should_return
NaN   */
NaN   for (uint256 i = 0; i < _operators.length; i++) {
NaN     require(addAddressToWhitelist(_operators[i]));
NaN   }
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 11: onlyOwner_addAddressesToWhitelist

08, Sep 2018

Posted by CTK report generator

1ms

Line 487-489 in File CelerCrowdsale.sol

```
NaN /*@CTK "onlyOwner"
NaN   @post msg.sender != owner -> __reverted
NaN */
```

Line 843-859 in File CelerCrowdsale.sol

```
NaN function addAddressesToWhitelist(address[] _operators)
NaN   onlyOwner
NaN   public
NaN   returns (bool)
NaN {
NaN   /*@CTK addAddressesToWhitelist_forloop
NaN     @inv i <= _operators.length
NaN     @inv _operators == _operators__pre
NaN     @inv forall j: uint. (j >= 0 /\ j < i) -> this.whitelisted[_operators[j]] == tr
NaN     @post i == _operators.length
NaN     @post !__should_return
NaN   */
NaN   for (uint256 i = 0; i < _operators.length; i++) {
NaN     require(addAddressToWhitelist(_operators[i]));
NaN   }
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 12: removeAddressFromWhitelist happy case

08, Sep 2018

Posted by CTK report generator

1.3ms

Line 867-871 in File CelerCrowdsale.sol

```
NaN /*@CTK "removeAddressFromWhitelist happy case"
NaN   @tag assume_completion
NaN   @post post(this).whitelisted[_operator] == false
NaN   @post __return == true
NaN */
```

Line 872-880 in File CelerCrowdsale.sol

```
NaN function removeAddressFromWhitelist(address _operator)
NaN   onlyOwner
NaN   public
NaN   returns (bool)
NaN {
NaN   whitelisted[_operator] = false;
NaN   emit WhitelistRemoved(_operator);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 13: onlyOwner_removeAddressFromWhitelist

08, Sep 2018

Posted by CTK report generator

0.7ms

Line 487-489 in File CelerCrowdsale.sol

```
NaN /*@CTK "onlyOwner"
NaN   @post msg.sender != owner -> __reverted
NaN */
```

Line 872-880 in File CelerCrowdsale.sol

```
NaN function removeAddressFromWhitelist(address _operator)
NaN   onlyOwner
NaN   public
NaN   returns (bool)
NaN {
NaN   whitelisted[_operator] = false;
NaN   emit WhitelistRemoved(_operator);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 14: removeAddressesFromWhitelist happy case

08, Sep 2018

Posted by CTK report generator

4.5ms

Line 888-892 in File CelerCrowdsale.sol

```
NaN /*@CTK "removeAddressesFromWhitelist happy case"
NaN   @tag assume_completion
NaN   @post forall i: uint. (i >= 0 /\ i < _operators.length) -> post(this).whitelistec
NaN   @post __return == true
NaN */
```

Line 893-909 in File CelerCrowdsale.sol

```
NaN function removeAddressesFromWhitelist(address[] _operators)
NaN   onlyOwner
NaN   public
NaN   returns (bool)
NaN {
NaN   /*@CTK removeAddressesFromWhitelist_forLoop
NaN     @inv i <= _operators.length
NaN     @inv _operators == _operators__pre
NaN     @inv forall j: uint. (j >= 0 /\ j < i) -> this.whitelisted[_operators[j]] == fa
NaN     @post i == _operators.length
NaN     @post !__should_return
NaN   */
NaN   for (uint256 i = 0; i < _operators.length; i++) {
NaN     require(removeAddressFromWhitelist(_operators[i]));
NaN   }
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 15: onlyOwner_removeAddressesFromWhitelist

08, Sep 2018

Posted by CTK report generator

1.2ms

Line 487-489 in File CelerCrowdsale.sol

```
NaN /*@CTK "onlyOwner"
NaN   @post msg.sender != owner -> __reverted
NaN */
```

Line 893-909 in File CelerCrowdsale.sol

```
NaN function removeAddressesFromWhitelist(address[] _operators)
NaN   onlyOwner
NaN   public
NaN   returns (bool)
NaN {
NaN   /*@CTK removeAddressesFromWhitelist_forLoop
NaN     @inv i <= _operators.length
NaN     @inv _operators == _operators__pre
NaN     @inv forall j: uint. (j >= 0 /\ j < i) -> this.whitelisted[_operators[j]] == fa
NaN     @post i == _operators.length
NaN     @post !__should_return
NaN   */
NaN   for (uint256 i = 0; i < _operators.length; i++) {
NaN     require(removeAddressFromWhitelist(_operators[i]));
NaN   }
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 16: pause

08, Sep 2018

Posted by CTK report generator

4.2ms

Line 578-581 in File CelerCrowdsale.sol

```
NaN /*@CTK "pause"
NaN   @tag assume_completion
NaN   @post post(this).paused == true
NaN */
```

Line 582-585 in File CelerCrowdsale.sol

```
NaN function pause() onlyOwner whenNotPaused public {
NaN   paused = true;
NaN   emit Pause();
NaN }
```

The code meets the specification

Detail for Request 17: onlyOwner_pause

08, Sep 2018

Posted by CTK report generator

2.6ms

Line 487-489 in File CelerCrowdsale.sol

```
NaN /*@CTK "onlyOwner"
NaN   @post msg.sender != owner -> __reverted
NaN */
```

Line 582-585 in File CelerCrowdsale.sol

```
NaN function pause() onlyOwner whenNotPaused public {
NaN   paused = true;
NaN   emit Pause();
NaN }
```

The code meets the specification

Detail for Request 18: whenNotPaused_pause

08, Sep 2018

Posted by CTK report generator

0.8ms

Line 556-558 in File CelerCrowdsale.sol

```
NaN /*@CTK "whenNotPaused"
NaN   @post this.paused -> __reverted
NaN */
```

Line 582-585 in File CelerCrowdsale.sol

```
NaN function pause() onlyOwner whenNotPaused public {
NaN   paused = true;
NaN   emit Pause();
NaN }
```

The code meets the specification

Detail for Request 19: unpause

08, Sep 2018

Posted by CTK report generator

2.1ms

Line 590-593 in File CelerCrowdsale.sol

```
NaN /*@CTK "unpause"
NaN   @tag assume_completion
NaN   @post post(this).paused == false
NaN */
```

Line 594-597 in File CelerCrowdsale.sol

```
NaN function unpause() onlyOwner whenPaused public {
NaN   paused = false;
NaN   emit Unpause();
NaN }
```

The code meets the specification

Detail for Request 20: onlyOwner_unpause

08, Sep 2018

Posted by CTK report generator

1.7ms

Line 487-489 in File CelerCrowdsale.sol

```
NaN /*@CTK "onlyOwner"
NaN   @post msg.sender != owner -> __reverted
NaN */
```

Line 594-597 in File CelerCrowdsale.sol

```
NaN function unpause() onlyOwner whenPaused public {
NaN   paused = false;
NaN   emit Unpause();
NaN }
```

The code meets the specification

Detail for Request 21: whenPaused_unpause

08, Sep 2018

Posted by CTK report generator

0.8ms

Line 567-569 in File CelerCrowdsale.sol

```
NaN /*@CTK "whenPaused"
NaN   @post !this.paused -> __reverted
NaN */
```

Line 594-597 in File CelerCrowdsale.sol

```
NaN function unpause() onlyOwner whenPaused public {
NaN   paused = false;
NaN   emit Unpause();
NaN }
```

The code meets the specification

Detail for Request 22: SafeMath_mul

08, Sep 2018

Posted by CTK report generator

282.1ms

Line 1119-1128 in File CelerCrowdsale.sol

```
NaN /*@CTK SafeMath_mul
NaN     @tag spec
NaN     @post __reverted == __has_assertion_failure
NaN     @post __has_assertion_failure == __has_overflow
NaN     @post __reverted == false -> c == _a * _b
NaN     @post _a == 0 -> c == 0
NaN     @post msg == msg__post
NaN     @post (_a > 0 && (_a * _b / _a != _b)) == __has_assertion_failure
NaN     @post __addr_map == __addr_map__post
NaN */
```

Line 1129-1140 in File CelerCrowdsale.sol

```
NaN function mul(uint256 _a, uint256 _b) internal pure returns (uint256 c) {
NaN   // Gas optimization: this is cheaper than asserting 'a' not being zero, but the
NaN   // benefit is lost if 'b' is also tested.
NaN   // See: https://github.com/OpenZeppelin/openzeppelin-solidity/pull/522
NaN   if (_a == 0) {
NaN     return 0;
NaN   }
NaN
NaN   c = _a * _b;
NaN   assert(c / _a == _b);
NaN   return c;
NaN }
```

The code meets the specification

Detail for Request 23: SafeMath_div

Posted by CTK report generator

08, Sep 2018
0.4ms

Line 1145-1153 in File CelerCrowdsale.sol

```
NaN /*@CTK SafeMath_div
NaN     @tag spec
NaN     @post __reverted == __has_assertion_failure
NaN     @post _b == 0 -> __reverted == true  // solidity throws on 0.
NaN     @post __has_overflow == true -> __has_assertion_failure == true
NaN     @post __reverted == false -> __return == _a / _b
NaN     @post msg == msg__post
NaN     @post __addr_map == __addr_map__post
NaN */
```

Line 1154-1159 in File CelerCrowdsale.sol

```
NaN function div(uint256 _a, uint256 _b) internal pure returns (uint256) {
NaN   // assert(_b > 0); // Solidity automatically throws when dividing by 0
NaN   // uint256 c = _a / _b;
NaN   // assert(_a == _b * c + _a % _b); // There is no case in which this doesn't holc
NaN   return _a / _b;
NaN }
```

The code meets the specification

Detail for Request 24: SafeMath_sub

08, Sep 2018

Posted by CTK report generator

1.2ms

Line 1164-1172 in File CelerCrowdsale.sol

```
NaN /*@CTK SafeMath_sub
NaN     @tag spec
NaN     @post __reverted == __has_assertion_failure
NaN     @post __has_overflow == true -> __has_assertion_failure == true
NaN     @post __reverted == false -> __return == _a - _b
NaN     @post msg == msg__post
NaN     @post (_a < _b) == __has_assertion_failure
NaN     @post __addr_map == __addr_map__post
NaN */
```

Line 1173-1176 in File CelerCrowdsale.sol

```
NaN function sub(uint256 _a, uint256 _b) internal pure returns (uint256) {
NaN   assert(_b <= _a);
NaN   return _a - _b;
NaN }
```

The code meets the specification

Detail for Request 25: SafeMath_add

08, Sep 2018

Posted by CTK report generator

3.1ms

Line 1181-1189 in File CelerCrowdsale.sol

```
NaN /*@CTK SafeMath_add
NaN     @tag spec
NaN     @post __reverted == __has_assertion_failure
NaN     @post __has_assertion_failure == __has_overflow
NaN     @post __reverted == false -> c == _a + _b
NaN     @post msg == msg__post
NaN     @post (_a + _b < _a)  == __has_assertion_failure
NaN     @post __addr_map == __addr_map__post
NaN */
```

Line 1190-1194 in File CelerCrowdsale.sol

```
NaN function add(uint256 _a, uint256 _b) internal pure returns (uint256 c) {
NaN   c = _a + _b;
NaN   assert(c >= _a);
NaN   return c;
NaN }
```

The code meets the specification

Detail for Request 26: If method completes, integer overflow would not happen.

08, Sep 2018

Posted by CTK report generator

0.8ms

Line 404 in File CelerCrowdsale.sol

```
NaN //@CTK NO_OVERFLOW
```

Line 410-412 in File CelerCrowdsale.sol

```
NaN function totalSupply() public view returns (uint256) {
NaN   return totalSupply_;
NaN }
```

The code meets the specification

Detail for Request 27: Method will not encounter an assertion failure.

08, Sep 2018

0.5ms

Posted by CTK report generator

Line 405 in File CelerCrowdsale.sol

```
NaN //@CTK NO_ASF
```

Line 410-412 in File CelerCrowdsale.sol

```
NaN function totalSupply() public view returns (uint256) {
NaN   return totalSupply_;
NaN }
```

The code meets the specification

Detail for Request 28: Buffer overflow / array index out of bound would never happen.

08, Sep 2018

0.6ms

Posted by CTK report generator

Line 406 in File CelerCrowdsale.sol

```
NaN //@CTK NO_BUF_OVERFLOW
```

Line 410-412 in File CelerCrowdsale.sol

```
NaN function totalSupply() public view returns (uint256) {
NaN   return totalSupply_;
NaN }
```

The code meets the specification

Detail for Request 29: token_total_supply

Posted by CTK report generator

08, Sep 2018
0.2ms

Line 407-409 in File CelerCrowdsale.sol

```
NaN /*@CTK token_total_supply
NaN   @post __return == this.totalSupply_
NaN */
```

Line 410-412 in File CelerCrowdsale.sol

```
NaN function totalSupply() public view returns (uint256) {
NaN   return totalSupply_;
NaN }
```

The code meets the specification

Detail for Request 30: If method completes, integer overflow would not happen.

08, Sep 2018

Posted by CTK report generator

12.5ms

Line 419 in File CelerCrowdsale.sol

```
NaN //@CTK NO_OVERFLOW
```

Line 432-440 in File CelerCrowdsale.sol

```
NaN function transfer(address _to, uint256 _value) public returns (bool) {
NaN   require(_to != address(0));
NaN   require(_value <= balances[msg.sender]);
NaN
NaN   balances[msg.sender] = balances[msg.sender].sub(_value);
NaN   balances[_to] = balances[_to].add(_value);
NaN   emit Transfer(msg.sender, _to, _value);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 31: transfer success case

08, Sep 2018

Posted by CTK report generator

153.3ms

Line 420-427 in File CelerCrowdsale.sol

```
NaN /*@CTK "transfer success case"
NaN   @tag assume_completion
NaN   @pre _to != address(0)
NaN   @pre balances[msg.sender] >= _value
NaN   @post _to != msg.sender -> __post.balances[msg.sender] == balances[msg.sender] -
NaN   @post _to != msg.sender -> __post.balances[_to] == balances[_to] + _value
NaN   @post __return == true
NaN */
```

Line 432-440 in File CelerCrowdsale.sol

```
NaN function transfer(address _to, uint256 _value) public returns (bool) {
NaN   require(_to != address(0));
NaN   require(_value <= balances[msg.sender]);
NaN
NaN   balances[msg.sender] = balances[msg.sender].sub(_value);
NaN   balances[_to] = balances[_to].add(_value);
NaN   emit Transfer(msg.sender, _to, _value);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 32: transfer reverted case

08, Sep 2018

Posted by CTK report generator

14.4ms

Line 428-431 in File CelerCrowdsale.sol

```
NaN /*@CTK "transfer reverted case"
NaN   @pre _to == address(0) \/ balances[msg.sender] < _value
NaN   @post __reverted == true
NaN */
```

Line 432-440 in File CelerCrowdsale.sol

```
NaN function transfer(address _to, uint256 _value) public returns (bool) {
NaN   require(_to != address(0));
NaN   require(_value <= balances[msg.sender]);
NaN
NaN   balances[msg.sender] = balances[msg.sender].sub(_value);
NaN   balances[_to] = balances[_to].add(_value);
NaN   emit Transfer(msg.sender, _to, _value);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 33: transfer success case

08, Sep 2018

Posted by CTK report generator

0.3ms

Line 447-449 in File CelerCrowdsale.sol

```
NaN /*@CTK "transfer success case"
NaN   @post __return == this.balances[_owner]
NaN */
```

Line 450-452 in File CelerCrowdsale.sol

```
NaN function balanceOf(address _owner) public view returns (uint256) {
NaN   return balances[_owner];
NaN }
```

The code meets the specification

Detail for Request 34: transferFrom success

08, Sep 2018

Posted by CTK report generator

204ms

Line 619-628 in File CelerCrowdsale.sol

```
NaN /*@CTK "transferFrom success"
NaN   @tag assume_completion
NaN   @pre _to != address(0)
NaN   @pre _value <= balances[_from]
NaN   @pre _value <= allowed[_from][msg.sender]
NaN   @post _from != _to -> __post.balances[_from] == balances[_from] - _value
NaN   @post _from != _to -> __post.balances[_to] == balances[_to] + _value
NaN   @post __post.allowed[_from][msg.sender] == allowed[_from][msg.sender] - _value
NaN   @post __return == true
NaN */
```

Line 641-658 in File CelerCrowdsale.sol

```
NaN function transferFrom(
NaN   address _from,
NaN   address _to,
NaN   uint256 _value
NaN )
NaN   public
NaN   returns (bool)
NaN {
NaN   require(_to != address(0));
NaN   require(_value <= balances[_from]);
```

```
NaN   require(_value <= allowed[_from][msg.sender]);
NaN
NaN   balances[_from] = balances[_from].sub(_value);
NaN   balances[_to] = balances[_to].add(_value);
NaN   allowed[_from][msg.sender] = allowed[_from][msg.sender].sub(_value);
NaN   emit Transfer(_from, _to, _value);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 35: transferFrom failure case 1: no enough balance

08, Sep 2018

Posted by CTK report generator

22ms

Line 629-632 in File CelerCrowdsale.sol

```
NaN /*@CTK "transferFrom failure case 1: no enough balance"
NaN   @pre balances[_from] < _value
NaN   @post __reverted
NaN */
```

Line 641-658 in File CelerCrowdsale.sol

```
NaN function transferFrom(
NaN   address _from,
NaN   address _to,
NaN   uint256 _value
NaN )
NaN   public
NaN   returns (bool)
NaN {
NaN   require(_to != address(0));
NaN   require(_value <= balances[_from]);
NaN   require(_value <= allowed[_from][msg.sender]);
NaN
NaN   balances[_from] = balances[_from].sub(_value);
NaN   balances[_to] = balances[_to].add(_value);
NaN   allowed[_from][msg.sender] = allowed[_from][msg.sender].sub(_value);
NaN   emit Transfer(_from, _to, _value);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 36: transferFrom failure case 2: no enough allowance

08, Sep 2018

Posted by CTK report generator

19.5ms

Line 633-636 in File CelerCrowdsale.sol

```
NaN /*@CTK "transferFrom failure case 2: no enough allowance"
NaN   @pre allowed[_from][msg.sender] < _value
NaN   @post __reverted
NaN */
```

Line 641-658 in File CelerCrowdsale.sol

```
NaN function transferFrom(
NaN   address _from,
NaN   address _to,
NaN   uint256 _value
NaN )
NaN   public
NaN   returns (bool)
NaN {
NaN   require(_to != address(0));
NaN   require(_value <= balances[_from]);
NaN   require(_value <= allowed[_from][msg.sender]);
NaN
NaN   balances[_from] = balances[_from].sub(_value);
NaN   balances[_to] = balances[_to].add(_value);
NaN   allowed[_from][msg.sender] = allowed[_from][msg.sender].sub(_value);
NaN   emit Transfer(_from, _to, _value);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 37: transferFrom failure case 3: _to is 0

08, Sep 2018

Posted by CTK report generator

0.9ms

Line 637-640 in File CelerCrowdsale.sol

```
NaN /*@CTK "transferFrom failure case 3: _to is 0"
NaN   @pre _to == address(0)
NaN   @post __reverted
NaN */
```

Line 641-658 in File CelerCrowdsale.sol

```
NaN function transferFrom(
NaN   address _from,
NaN   address _to,
```

```
NaN   uint256 _value
NaN )
NaN   public
NaN   returns (bool)
NaN {
NaN   require(_to != address(0));
NaN   require(_value <= balances[_from]);
NaN   require(_value <= allowed[_from][msg.sender]);
NaN
NaN   balances[_from] = balances[_from].sub(_value);
NaN   balances[_to] = balances[_to].add(_value);
NaN   allowed[_from][msg.sender] = allowed[_from][msg.sender].sub(_value);
NaN   emit Transfer(_from, _to, _value);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 38: If method completes, integer overflow would not happen.

08, Sep 2018

Posted by CTK report generator

0.5ms

Line 670 in File CelerCrowdsale.sol

```
NaN //@CTK NO_OVERFLOW
```

Line 676-680 in File CelerCrowdsale.sol

```
NaN function approve(address _spender, uint256 _value) public returns (bool) {
NaN   allowed[msg.sender][_spender] = _value;
NaN   emit Approval(msg.sender, _spender, _value);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 39: Method will not encounter an assertion failure.

08, Sep 2018

Posted by CTK report generator

0.7ms

Line 671 in File CelerCrowdsale.sol

```
NaN //@CTK NO_ASF
```

Line 676-680 in File CelerCrowdsale.sol

```
NaN function approve(address _spender, uint256 _value) public returns (bool) {
NaN   allowed[msg.sender][_spender] = _value;
NaN   emit Approval(msg.sender, _spender, _value);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 40: approve transfer allowance

08, Sep 2018

Posted by CTK report generator

2.5ms

Line 672-675 in File CelerCrowdsale.sol

```
NaN /*@CTK "approve transfer allowance"
NaN   @post post(this).allowed[msg.sender][_spender] == _value
NaN   @post __return == true
NaN */
```

Line 676-680 in File CelerCrowdsale.sol

```
NaN function approve(address _spender, uint256 _value) public returns (bool) {
NaN   allowed[msg.sender][_spender] = _value;
NaN   emit Approval(msg.sender, _spender, _value);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 41: get the allowance

08, Sep 2018

Posted by CTK report generator

0.5ms

Line 688-691 in File CelerCrowdsale.sol

```
NaN /*@CTK "get the allowance"
NaN   @post __return == allowed[_owner][_spender]
NaN   @post this == post(this)
NaN */
```

Line 692-701 in File CelerCrowdsale.sol

```
NaN function allowance(
NaN   address _owner,
NaN   address _spender
NaN  )
NaN   public
NaN   view
NaN   returns (uint256)
NaN {
NaN   return allowed[_owner][_spender];
NaN }
```

The code meets the specification

Detail for Request 42: increaseApproval ok

08, Sep 2018

Posted by CTK report generator

1.9ms

Line 712-716 in File CelerCrowdsale.sol

```
NaN /*@CTK "increaseApproval ok"
NaN   @tag assume_completion
NaN   @post post(this).allowed[msg.sender][_spender] == this.allowed[msg.sender][_spenc
NaN   @post __return == true
NaN */
```

Line 717-728 in File CelerCrowdsale.sol

```
NaN function increaseApproval(
NaN   address _spender,
NaN   uint256 _addedValue
NaN )
NaN   public
NaN   returns (bool)
NaN {
NaN   allowed[msg.sender][_spender] = (
NaN     allowed[msg.sender][_spender].add(_addedValue));
NaN   emit Approval(msg.sender, _spender, allowed[msg.sender][_spender]);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 43: decreaseApproval case if

08, Sep 2018

Posted by CTK report generator

2.1ms

Line 739-744 in File CelerCrowdsale.sol

```
NaN /*@CTK "decreaseApproval case if"
NaN   @tag assume_completion
NaN   @pre _subtractedValue > allowed[msg.sender][_spender]
NaN   @post post(this).allowed[msg.sender][_spender] == 0
NaN   @post __return == true
NaN */
```

Line 751-766 in File CelerCrowdsale.sol

```
NaN function decreaseApproval(
NaN   address _spender,
NaN   uint256 _subtractedValue
NaN )
NaN   public
NaN   returns (bool)
NaN {
NaN   uint256 oldValue = allowed[msg.sender][_spender];
NaN   if (_subtractedValue > oldValue) {
NaN     allowed[msg.sender][_spender] = 0;
NaN   } else {
NaN     allowed[msg.sender][_spender] = oldValue.sub(_subtractedValue);
NaN   }
NaN   emit Approval(msg.sender, _spender, allowed[msg.sender][_spender]);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 44: decreaseApproval case else

08, Sep 2018

Posted by CTK report generator

3.1ms

Line 745-750 in File CelerCrowdsale.sol

```
NaN /*@CTK "decreaseApproval case else"
NaN   @tag assume_completion
NaN   @pre _subtractedValue <= allowed[msg.sender][_spender]
NaN   @post post(this).allowed[msg.sender][_spender] == this.allowed[msg.sender][_spenc
NaN   @post __return == true
NaN */
```

Line 751-766 in File CelerCrowdsale.sol

```
NaN function decreaseApproval(
NaN   address _spender,
NaN   uint256 _subtractedValue
NaN )
NaN   public
NaN   returns (bool)
```

```
NaN {
NaN   uint256 oldValue = allowed[msg.sender][_spender];
NaN   if (_subtractedValue > oldValue) {
NaN     allowed[msg.sender][_spender] = 0;
NaN   } else {
NaN     allowed[msg.sender][_spender] = oldValue.sub(_subtractedValue);
NaN   }
NaN   emit Approval(msg.sender, _spender, allowed[msg.sender][_spender]);
NaN   return true;
NaN }
```

The code meets the specification

Detail for Request 45: PausableToken transfer success case

08, Sep 2018

Posted by CTK report generator

160.6ms

Line 920-925 in File CelerCrowdsale.sol

```
NaN /*@CTK "PausableToken transfer success case"
NaN   @tag assume_completion
NaN   @post _to != msg.sender -> __post.balances[msg.sender] == balances[msg.sender] -
NaN   @post _to != msg.sender -> __post.balances[_to] == balances[_to] + _value
NaN   @post __return == true
NaN */
```

Line 926-935 in File CelerCrowdsale.sol

```
NaN function transfer(
NaN   address _to,
NaN   uint256 _value
NaN )
NaN   public
NaN   whenNotPaused
NaN   returns (bool)
NaN {
NaN   return super.transfer(_to, _value);
NaN }
```

The code meets the specification

Detail for Request 46: whenNotPaused_transfer

08, Sep 2018

Posted by CTK report generator

2.1ms

Line 556-558 in File CelerCrowdsale.sol

```
NaN /*@CTK "whenNotPaused"
NaN   @post this.paused -> __reverted
NaN */
```

Line 926-935 in File CelerCrowdsale.sol

```
NaN function transfer(
NaN   address _to,
NaN   uint256 _value
NaN )
NaN   public
NaN   whenNotPaused
NaN   returns (bool)
NaN {
NaN   return super.transfer(_to, _value);
NaN }
```

The code meets the specification

Detail for Request 47: PausableToken transferFrom success

08, Sep 2018

Posted by CTK report generator

285.7ms

Line 937-943 in File CelerCrowdsale.sol

```
NaN /*@CTK "PausableToken transferFrom success"
NaN   @tag assume_completion
NaN   @post _from != _to -> __post.balances[_from] == balances[_from] - _value
NaN   @post _from != _to -> __post.balances[_to] == balances[_to] + _value
NaN   @post __post.allowed[_from][msg.sender] == allowed[_from][msg.sender] - _value
NaN   @post __return == true
NaN */
```

Line 944-954 in File CelerCrowdsale.sol

```
NaN function transferFrom(
NaN   address _from,
NaN   address _to,
NaN   uint256 _value
NaN )
NaN   public
NaN   whenNotPaused
NaN   returns (bool)
NaN {
NaN   return super.transferFrom(_from, _to, _value);
NaN }
```

The code meets the specification

Detail for Request 48: whenNotPaused_transferFrom

08, Sep 2018

Posted by CTK report generator

1.6ms

Line 556-558 in File CelerCrowdsale.sol

```
NaN /*@CTK "whenNotPaused"
NaN   @post this.paused -> __reverted
NaN */
```

Line 944-954 in File CelerCrowdsale.sol

```
NaN function transferFrom(
NaN   address _from,
NaN   address _to,
NaN   uint256 _value
NaN )
NaN   public
NaN   whenNotPaused
NaN   returns (bool)
NaN {
NaN   return super.transferFrom(_from, _to, _value);
NaN }
```

The code meets the specification

Detail for Request 49: PausableToken approve transfer allowance

08, Sep 2018

Posted by CTK report generator

3.4ms

Line 956-960 in File CelerCrowdsale.sol

```
NaN /*@CTK "PausableToken approve transfer allowance"
NaN   @pre paused == false
NaN   @post post(this).allowed[msg.sender][_spender] == _value
NaN   @post __return == true
NaN */
```

Line 961-970 in File CelerCrowdsale.sol

```
NaN function approve(
NaN   address _spender,
NaN   uint256 _value
NaN )
NaN   public
NaN   whenNotPaused
```

```
NaN   returns (bool)
NaN {
NaN   return super.approve(_spender, _value);
NaN }
```

The code meets the specification

Detail for Request 50: whenNotPaused_approve

08, Sep 2018

Posted by CTK report generator

1.1ms

Line 556-558 in File CelerCrowdsale.sol

```
NaN /*@CTK "whenNotPaused"
NaN   @post this.paused -> __reverted
NaN */
```

Line 961-970 in File CelerCrowdsale.sol

```
NaN function approve(
NaN   address _spender,
NaN   uint256 _value
NaN )
NaN   public
NaN   whenNotPaused
NaN   returns (bool)
NaN {
NaN   return super.approve(_spender, _value);
NaN }
```

The code meets the specification

Detail for Request 51: increaseApproval ok

08, Sep 2018

Posted by CTK report generator

3.7ms

Line 971-974 in File CelerCrowdsale.sol

```
NaN /*@CTK "increaseApproval ok"
NaN   @tag assume_completion
NaN   @post post(this).allowed[msg.sender][_spender] == this.allowed[msg.sender][_spenc
NaN */
```

Line 975-984 in File CelerCrowdsale.sol

```
NaN function increaseApproval(
NaN   address _spender,
NaN   uint _addedValue
NaN )
NaN   public
NaN   whenNotPaused
NaN   returns (bool success)
NaN {
NaN   return super.increaseApproval(_spender, _addedValue);
NaN }
```

The code meets the specification

Detail for Request 52: whenNotPaused_increaseApproval

08, Sep 2018

Posted by CTK report generator

1.2ms

Line 556-558 in File CelerCrowdsale.sol

```
NaN /*@CTK "whenNotPaused"
NaN   @post this.paused -> __reverted
NaN */
```

Line 975-984 in File CelerCrowdsale.sol

```
NaN function increaseApproval(
NaN   address _spender,
NaN   uint _addedValue
NaN )
NaN   public
NaN   whenNotPaused
NaN   returns (bool success)
NaN {
NaN   return super.increaseApproval(_spender, _addedValue);
NaN }
```

The code meets the specification

Detail for Request 53: decreaseApproval case if

08, Sep 2018

Posted by CTK report generator

25.8ms

Line 986-990 in File CelerCrowdsale.sol

```
NaN /*@CTK "decreaseApproval case if"
NaN   @tag assume_completion
NaN   @pre _subtractedValue > allowed[msg.sender][_spender]
NaN   @post post(this).allowed[msg.sender][_spender] == 0
NaN */
```

Line 996-1005 in File CelerCrowdsale.sol

```
NaN function decreaseApproval(
NaN   address _spender,
NaN   uint _subtractedValue
NaN )
NaN   public
NaN   whenNotPaused
NaN   returns (bool success)
NaN {
NaN   return super.decreaseApproval(_spender, _subtractedValue);
NaN }
```

The code meets the specification

Detail for Request 54: decreaseApproval case else

08, Sep 2018

Posted by CTK report generator

47.1ms

Line 991-995 in File CelerCrowdsale.sol

```
NaN /*@CTK "decreaseApproval case else"
NaN   @tag assume_completion
NaN   @pre _subtractedValue <= allowed[msg.sender][_spender]
NaN   @post post(this).allowed[msg.sender][_spender] == this.allowed[msg.sender][_spenc
NaN */
```

Line 996-1005 in File CelerCrowdsale.sol

```
NaN function decreaseApproval(
NaN   address _spender,
NaN   uint _subtractedValue
NaN )
NaN   public
NaN   whenNotPaused
NaN   returns (bool success)
NaN {
NaN   return super.decreaseApproval(_spender, _subtractedValue);
NaN }
```

The code meets the specification

Detail for Request 55: whenNotPaused_decreaseApproval

08, Sep 2018

Posted by CTK report generator

1.4ms

Line 556-558 in File CelerCrowdsale.sol

```
NaN /*@CTK "whenNotPaused"
NaN   @post this.paused -> __reverted
NaN */
```

Line 996-1005 in File CelerCrowdsale.sol

```
NaN function decreaseApproval(
NaN   address _spender,
NaN   uint _subtractedValue
NaN )
NaN   public
NaN   whenNotPaused
NaN   returns (bool success)
NaN {
NaN   return super.decreaseApproval(_spender, _subtractedValue);
NaN }
```

The code meets the specification

Detail for Request 56: CelerToken constructor

08, Sep 2018

Posted by CTK report generator

2.7ms

Line 1042-1045 in File CelerCrowdsale.sol

```
NaN /*@CTK "CelerToken constructor"
NaN   @post post(this).totalSupply_ == INITIAL_SUPPLY
NaN   @post post(this).balances[msg.sender] == INITIAL_SUPPLY
NaN */
```

Line 1046-1049 in File CelerCrowdsale.sol

```
NaN constructor() public {
NaN   totalSupply_ = INITIAL_SUPPLY;
NaN   balances[msg.sender] = INITIAL_SUPPLY;
NaN }
```

The code meets the specification

Detail for Request 57: CelerToken transfer success case

08, Sep 2018

Posted by CTK report generator

223.2ms

Line 1055-1060 in File CelerCrowdsale.sol

```
NaN /*@CTK "CelerToken transfer success case"
NaN   @tag assume_completion
NaN   @post _to != msg.sender -> __post.balances[msg.sender] == balances[msg.sender] -
NaN   @post _to != msg.sender -> __post.balances[_to] == balances[_to] + _value
NaN   @post __return == true
NaN */
```

Line 1061-1071 in File CelerCrowdsale.sol

```
NaN function transfer(
NaN   address _to,
NaN   uint256 _value
NaN )
NaN   public
NaN   onlyIfTransferable
NaN   onlyValidReceiver(_to)
NaN   returns (bool)
NaN {
NaN   return super.transfer(_to, _value);
NaN }
```

The code meets the specification

Detail for Request 58: onlyValidReceiver_transfer

08, Sep 2018

Posted by CTK report generator

7ms

Line 1024-1027 in File CelerCrowdsale.sol

```
NaN /*@CTK onlyValidReceiver
NaN   @pre this.transferOpened == false /\ whitelisted[msg.sender] == false /\ msg.senc
NaN   @post __reverted
NaN */
```

Line 1061-1071 in File CelerCrowdsale.sol

```
NaN function transfer(
NaN   address _to,
NaN   uint256 _value
NaN )
NaN   public
NaN   onlyIfTransferable
NaN   onlyValidReceiver(_to)
NaN   returns (bool)
```

```
NaN {
NaN   return super.transfer(_to, _value);
NaN }
```

The code meets the specification

Detail for Request 59: onlyValidReceiver_transfer

08, Sep 2018

Posted by CTK report generator

7ms

Line 1033-1035 in File CelerCrowdsale.sol

```
NaN /*@CTK onlyValidReceiver
NaN   @post _to == address(this) -> __reverted
NaN */
```

Line 1061-1071 in File CelerCrowdsale.sol

```
NaN function transfer(
NaN   address _to,
NaN   uint256 _value
NaN )
NaN   public
NaN   onlyIfTransferable
NaN   onlyValidReceiver(_to)
NaN   returns (bool)
NaN {
NaN   return super.transfer(_to, _value);
NaN }
```

The code meets the specification

Detail for Request 60: transferFrom transferFrom success

08, Sep 2018

Posted by CTK report generator

319.5ms

Line 1077-1083 in File CelerCrowdsale.sol

```
NaN /*@CTK "transferFrom transferFrom success"
NaN   @tag assume_completion
NaN   @post _from != _to -> __post.balances[_from] == balances[_from] - _value
NaN   @post _from != _to -> __post.balances[_to] == balances[_to] + _value
NaN   @post __post.allowed[_from][msg.sender] == allowed[_from][msg.sender] - _value
```

```
NaN   @post __return == true
NaN */
```

Line 1084-1095 in File CelerCrowdsale.sol

```
NaN function transferFrom(
NaN   address _from,
NaN   address _to,
NaN   uint256 _value
NaN )
NaN   public
NaN   onlyIfTransferable
NaN   onlyValidReceiver(_to)
NaN   returns (bool)
NaN {
NaN   return super.transferFrom(_from, _to, _value);
NaN }
```

The code meets the specification

Detail for Request 61: onlyValidReceiver_transferFrom

08, Sep 2018

Posted by CTK report generator

9.7ms

Line 1024-1027 in File CelerCrowdsale.sol

```
NaN /*@CTK onlyValidReceiver
NaN   @pre this.transferOpened == false /\ whitelisted[msg.sender] == false /\ msg.senc
NaN   @post __reverted
NaN */
```

Line 1084-1095 in File CelerCrowdsale.sol

```
NaN function transferFrom(
NaN   address _from,
NaN   address _to,
NaN   uint256 _value
NaN )
NaN   public
NaN   onlyIfTransferable
NaN   onlyValidReceiver(_to)
NaN   returns (bool)
NaN {
NaN   return super.transferFrom(_from, _to, _value);
NaN }
```

The code meets the specification

Detail for Request 62: onlyValidReceiver_transferFrom

08, Sep 2018

Posted by CTK report generator

9.7ms

Line 1033-1035 in File CelerCrowdsale.sol

```
NaN /*@CTK onlyValidReceiver
NaN   @post _to == address(this) -> __reverted
NaN */
```

Line 1084-1095 in File CelerCrowdsale.sol

```
NaN function transferFrom(
NaN   address _from,
NaN   address _to,
NaN   uint256 _value
NaN )
NaN   public
NaN   onlyIfTransferable
NaN   onlyValidReceiver(_to)
NaN   returns (bool)
NaN {
NaN   return super.transferFrom(_from, _to, _value);
NaN }
```

The code meets the specification

Detail for Request 63: openTransfer ok

08, Sep 2018

Posted by CTK report generator

2.6ms

Line 1101-1104 in File CelerCrowdsale.sol

```
NaN /*@CTK "openTransfer ok"
NaN   @tag assume_completion
NaN   @post post(this).transferOpened == true
NaN */
```

Line 1105-1107 in File CelerCrowdsale.sol

```
NaN function openTransfer() external onlyOwner {
NaN   transferOpened = true;
NaN }
```

The code meets the specification

Detail for Request 64: onlyOwner_openTransfer

08, Sep 2018

Posted by CTK report generator

1.6ms

Line 487-489 in File CelerCrowdsale.sol

```
NaN /*@CTK "onlyOwner"
NaN   @post msg.sender != owner -> __reverted
NaN */
```

Line 1105-1107 in File CelerCrowdsale.sol

```
NaN function openTransfer() external onlyOwner {
NaN   transferOpened = true;
NaN }
```

The code meets the specification

Detail for Request 65: buyTokens happy case

08, Sep 2018

Posted by CTK report generator

26.6ms

Line 1357-1367 in File CelerCrowdsale.sol

```
NaN /*@CTK "buyTokens happy case"
NaN   @tag assume_completion
NaN   @pre _beneficiary != address(0)
NaN   @pre msg.value != 0
NaN   @pre token.balances[msg.sender] >= msg.value * this.rate
NaN   @post _beneficiary != msg.sender -> post(token).balances[msg.sender] == token.bal
NaN   @post _beneficiary != msg.sender -> post(token).balances[_beneficiary] == token.b
NaN   @post post(this).weiRaised == this.weiRaised + msg.value
NaN   @post post(this).wallet.balance == this.wallet.balance + msg.value
NaN   @post post(this).rate == this.rate
NaN  */
```

Line 1376-1399 in File CelerCrowdsale.sol

```
NaN function buyTokens(address _beneficiary) public payable {
NaN
NaN   uint256 weiAmount = msg.value;
NaN   _preValidatePurchase(_beneficiary, weiAmount);
NaN
NaN   // calculate token amount to be created
NaN   uint256 tokens = _getTokenAmount(weiAmount);
NaN
NaN   // update state
NaN   weiRaised = weiRaised.add(weiAmount);
NaN
NaN   _processPurchase(_beneficiary, tokens);
NaN   emit TokenPurchase(
```

```
NaN     msg.sender,
NaN     _beneficiary,
NaN     weiAmount,
NaN     tokens
NaN   );
NaN
NaN   _updatePurchasingState(_beneficiary, weiAmount);
NaN
NaN   _forwardFunds();
NaN   _postValidatePurchase(_beneficiary, weiAmount);
NaN }
```

The code meets the specification

Detail for Request 66: buyTokens reverted due to msg.value

08, Sep 2018

Posted by CTK report generator

13.4ms

Line 1368-1371 in File CelerCrowdsale.sol

```
NaN /*@CTK "buyTokens reverted due to msg.value"
NaN   @pre msg.value == 0
NaN   @post __reverted == true
NaN  */
```

Line 1376-1399 in File CelerCrowdsale.sol

```
NaN function buyTokens(address _beneficiary) public payable {
NaN
NaN   uint256 weiAmount = msg.value;
NaN   _preValidatePurchase(_beneficiary, weiAmount);
NaN
NaN   // calculate token amount to be created
NaN   uint256 tokens = _getTokenAmount(weiAmount);
NaN
NaN   // update state
NaN   weiRaised = weiRaised.add(weiAmount);
NaN
NaN   _processPurchase(_beneficiary, tokens);
NaN   emit TokenPurchase(
NaN     msg.sender,
NaN     _beneficiary,
NaN     weiAmount,
NaN     tokens
NaN   );
NaN
NaN   _updatePurchasingState(_beneficiary, weiAmount);
NaN
NaN   _forwardFunds();
NaN   _postValidatePurchase(_beneficiary, weiAmount);
NaN }
```

The code meets the specification

Detail for Request 67: buyTokens reverted due to _beneficiary

08, Sep 2018

Posted by CTK report generator

2.2ms

Line 1372-1375 in File CelerCrowdsale.sol

```
NaN /*@CTK "buyTokens reverted due to _beneficiary"
NaN   @pre _beneficiary == address(0)
NaN   @post __reverted == true
NaN  */
```

Line 1376-1399 in File CelerCrowdsale.sol

```
NaN function buyTokens(address _beneficiary) public payable {
NaN
NaN   uint256 weiAmount = msg.value;
NaN   _preValidatePurchase(_beneficiary, weiAmount);
NaN
NaN   // calculate token amount to be created
NaN   uint256 tokens = _getTokenAmount(weiAmount);
NaN
NaN   // update state
NaN   weiRaised = weiRaised.add(weiAmount);
NaN
NaN   _processPurchase(_beneficiary, tokens);
NaN   emit TokenPurchase(
NaN     msg.sender,
NaN     _beneficiary,
NaN     weiAmount,
NaN     tokens
NaN   );
NaN
NaN   _updatePurchasingState(_beneficiary, weiAmount);
NaN
NaN   _forwardFunds();
NaN   _postValidatePurchase(_beneficiary, weiAmount);
NaN }
```

The code meets the specification

Detail for Request 68: Validate purchase non reverted case

08, Sep 2018

Posted by CTK report generator

1.9ms

Line 1413-1417 in File CelerCrowdsale.sol

```
NaN /*@CTK "Validate purchase non reverted case"
NaN   @pre _beneficiary != address(0)
NaN   @pre _weiAmount != 0
NaN   @post !__reverted
NaN  */
```

Line 1426-1434 in File CelerCrowdsale.sol

```
NaN function _preValidatePurchase(
NaN   address _beneficiary,
NaN   uint256 _weiAmount
NaN )
NaN   internal
NaN {
NaN   require(_beneficiary != address(0));
NaN   require(_weiAmount != 0);
NaN }
```

The code meets the specification

Detail for Request 69: Validate purchase reverted on _beneficiary

08, Sep 2018

Posted by CTK report generator

0.3ms

Line 1418-1421 in File CelerCrowdsale.sol

```
NaN /*@CTK "Validate purchase reverted on _beneficiary"
NaN   @pre _beneficiary == address(0)
NaN   @post __reverted
NaN  */
```

Line 1426-1434 in File CelerCrowdsale.sol

```
NaN function _preValidatePurchase(
NaN   address _beneficiary,
NaN   uint256 _weiAmount
NaN )
NaN   internal
NaN {
NaN   require(_beneficiary != address(0));
NaN   require(_weiAmount != 0);
NaN }
```

The code meets the specification

Detail for Request 70: Validate purchase reverted on _weiAmount

08, Sep 2018

Posted by CTK report generator

2ms

Line 1422-1425 in File CelerCrowdsale.sol

```
NaN /*@CTK "Validate purchase reverted on _weiAmount"
NaN   @pre _weiAmount == 0
NaN   @post __reverted
NaN  */
```

Line 1426-1434 in File CelerCrowdsale.sol

```
NaN function _preValidatePurchase(
NaN   address _beneficiary,
NaN   uint256 _weiAmount
NaN )
NaN   internal
NaN {
NaN   require(_beneficiary != address(0));
NaN   require(_weiAmount != 0);
NaN }
```

The code meets the specification

Detail for Request 71: _postValidatePurchase should not change any state

08, Sep 2018

Posted by CTK report generator

0.2ms

Line 1441-1443 in File CelerCrowdsale.sol

```
NaN /*@CTK "_postValidatePurchase should not change any state"
NaN   @post post(this) == this
NaN  */
```

Line 1444-1451 in File CelerCrowdsale.sol

```
NaN function _postValidatePurchase(
NaN   address _beneficiary,
NaN   uint256 _weiAmount
NaN )
NaN   internal
NaN {
NaN   // optional override
NaN }
```

The code meets the specification

Detail for Request 72: _deliverTokens success case

08, Sep 2018

Posted by CTK report generator

112.8ms

Line 1458-1464 in File CelerCrowdsale.sol

```
NaN /*@CTK "_deliverTokens success case"
NaN   @tag assume_completion
NaN   @pre _beneficiary != address(0)
NaN   @pre token.balances[msg.sender] >= _tokenAmount
NaN   @post _beneficiary != msg.sender -> post(token).balances[msg.sender] == token.bal
NaN   @post _beneficiary != msg.sender -> post(token).balances[_beneficiary] == token.b
NaN */
```

Line 1465-1472 in File CelerCrowdsale.sol

```
NaN function _deliverTokens(
NaN   address _beneficiary,
NaN   uint256 _tokenAmount
NaN )
NaN   internal
NaN {
NaN   token.safeTransfer(_beneficiary, _tokenAmount);
NaN }
```

The code meets the specification

Detail for Request 73: _processPurchase success case

08, Sep 2018

Posted by CTK report generator

69.8ms

Line 1479-1485 in File CelerCrowdsale.sol

```
NaN /*@CTK "_processPurchase success case"
NaN   @tag assume_completion
NaN   @pre _beneficiary != address(0)
NaN   @pre token.balances[msg.sender] >= _tokenAmount
NaN   @post _beneficiary != msg.sender -> post(token).balances[msg.sender] == token.bal
NaN   @post _beneficiary != msg.sender -> post(token).balances[_beneficiary] == token.b
NaN */
```

Line 1486-1493 in File CelerCrowdsale.sol

```
NaN function _processPurchase(
```

```
NaN   address _beneficiary,
NaN   uint256 _tokenAmount
NaN )
NaN   internal
NaN {
NaN   _deliverTokens(_beneficiary, _tokenAmount);
NaN }
```

The code meets the specification

Detail for Request 74: _updatePurchasingState should not change any state

08, Sep 2018

Posted by CTK report generator

0.2ms

Line 1500-1502 in File CelerCrowdsale.sol

```
NaN /*@CTK "_updatePurchasingState should not change any state"
NaN   @post post(this) == this
NaN  */
```

Line 1503-1510 in File CelerCrowdsale.sol

```
NaN function _updatePurchasingState(
NaN   address _beneficiary,
NaN   uint256 _weiAmount
NaN )
NaN   internal
NaN {
NaN   // optional override
NaN }
```

The code meets the specification

Detail for Request 75: Get token amount

08, Sep 2018

Posted by CTK report generator

0.9ms

Line 1517-1520 in File CelerCrowdsale.sol

```
NaN /*@CTK "Get token amount"
NaN   @tag assume_completion
```

```
NaN   @post __return == _weiAmount * rate
NaN  */
```

Line 1521-1525 in File CelerCrowdsale.sol

```
NaN function _getTokenAmount(uint256 _weiAmount)
NaN   internal view returns (uint256)
NaN {
NaN   return _weiAmount.mul(rate);
NaN }
```

The code meets the specification

Detail for Request 76: Revert when transferring 0 amount of fund.

08, Sep 2018

Posted by CTK report generator

0.3ms

Line 1530-1533 in File CelerCrowdsale.sol

```
NaN /*@CTK "Revert when transferring 0 amount of fund."
NaN   @pre msg.value == 0
NaN   @post __reverted == true
NaN  */
```

Line 1537-1539 in File CelerCrowdsale.sol

```
NaN function _forwardFunds() internal {
NaN   wallet.transfer(msg.value);
NaN }
```

The code meets the specification

Detail for Request 77: _forwardFunds should not change any state

08, Sep 2018

Posted by CTK report generator

0.2ms

Line 1534-1536 in File CelerCrowdsale.sol

```
NaN /*@CTK "_forwardFunds should not change any state"
NaN   @post post(this) == this
NaN  */
```

Line 1537-1539 in File CelerCrowdsale.sol

```
NaN function _forwardFunds() internal {
NaN   wallet.transfer(msg.value);
NaN }
```

The code meets the specification

Detail for Request 78: AllowanceCrowdsale constructor happy case.

08, Sep 2018

Posted by CTK report generator

2ms

Line 1556-1559 in File CelerCrowdsale.sol

```
NaN /*@CTK "AllowanceCrowdsale constructor happy case."
NaN   @tag assume_completion
NaN   @post post(this).tokenWallet == _tokenWallet /\ _tokenWallet != address(0)
NaN */
```

Line 1564-1567 in File CelerCrowdsale.sol

```
NaN constructor(address _tokenWallet) public {
NaN   require(_tokenWallet != address(0));
NaN   tokenWallet = _tokenWallet;
NaN }
```

The code meets the specification

Detail for Request 79: AllowanceCrowdsale constructor exception case.

08, Sep 2018

Posted by CTK report generator

0.3ms

Line 1560-1563 in File CelerCrowdsale.sol

```
NaN /*@CTK "AllowanceCrowdsale constructor exception case."
NaN   @pre _tokenWallet == address(0)
NaN   @post post(__reverted) == true
NaN */
```

Line 1564-1567 in File CelerCrowdsale.sol

```
NaN constructor(address _tokenWallet) public {
NaN   require(_tokenWallet != address(0));
NaN   tokenWallet = _tokenWallet;
NaN }
```

The code meets the specification

Detail for Request 80: get the allowance

08, Sep 2018

Posted by CTK report generator

0.4ms

Line 1574-1577 in File CelerCrowdsale.sol

```
NaN /*@CTK "get the allowance"
NaN   @post __return == this.token.allowed[tokenWallet][this]
NaN   @post this == post(this)
NaN */
```

Line 1578-1582 in File CelerCrowdsale.sol

```
NaN function remainingTokens() public view returns (uint256) {
NaN   return token.allowance(tokenWallet, this);
NaN }
```

The code meets the specification

Detail for Request 81: _deliverTokens success

08, Sep 2018

Posted by CTK report generator

226.9ms

Line 1589-1597 in File CelerCrowdsale.sol

```
NaN /*@CTK "_deliverTokens success"
NaN     @tag assume_completion
NaN     @pre _beneficiary != address(0)
NaN     @pre _tokenAmount <= token.balances[tokenWallet]
NaN     @pre _tokenAmount <= token.allowed[tokenWallet][msg.sender]
NaN     @post tokenWallet != _beneficiary -> post(token).balances[tokenWallet] == toker
NaN     @post tokenWallet != _beneficiary -> post(token).balances[_beneficiary] == toke
NaN     @post post(token).allowed[tokenWallet][msg.sender] == token.allowed[tokenWallet
NaN */
```

Line 1598-1605 in File CelerCrowdsale.sol

```
NaN function _deliverTokens(
NaN   address _beneficiary,
NaN   uint256 _tokenAmount
```

```
NaN )
NaN   internal
NaN {
NaN   token.safeTransferFrom(tokenWallet, _beneficiary, _tokenAmount);
NaN }
```

The code meets the specification

Detail for Request 82: Validate _preValidatePurchase reverted on _beneficiary succinct

08, Sep 2018

Posted by CTK report generator

3.3ms

Line 1620-1623 in File CelerCrowdsale.sol

```
NaN /*@CTK "Validate _preValidatePurchase reverted on _beneficiary succinct"
NaN   @pre _beneficiary == address(0)
NaN   @post __reverted
NaN  */
```

Line 1632-1640 in File CelerCrowdsale.sol

```
NaN function _preValidatePurchase(
NaN   address _beneficiary,
NaN   uint256 _weiAmount
NaN )
NaN   onlyIfWhitelisted(_beneficiary)
NaN   internal
NaN {
NaN   super._preValidatePurchase(_beneficiary, _weiAmount);
NaN }
```

The code meets the specification

Detail for Request 83: Validate _preValidatePurchase reverted on _weiAmount succinct

08, Sep 2018

Posted by CTK report generator

4.5ms

Line 1624-1627 in File CelerCrowdsale.sol

```
NaN /*@CTK "Validate _preValidatePurchase reverted on _weiAmount succinct"
```

```
NaN   @pre _weiAmount == 0
NaN   @post __reverted
NaN  */
```

Line 1632-1640 in File CelerCrowdsale.sol

```
NaN function _preValidatePurchase(
NaN   address _beneficiary,
NaN   uint256 _weiAmount
NaN )
NaN   onlyIfWhitelisted(_beneficiary)
NaN   internal
NaN {
NaN   super._preValidatePurchase(_beneficiary, _weiAmount);
NaN }
```

The code meets the specification

Detail for Request 84: Validate _preValidatePurchase reverted on whitelist

08, Sep 2018

Posted by CTK report generator

1.6ms

Line 1628-1631 in File CelerCrowdsale.sol

```
NaN /*@CTK "Validate _preValidatePurchase reverted on whitelist"
NaN   @pre !whitelisted[_beneficiary]
NaN   @post __reverted
NaN  */
```

Line 1632-1640 in File CelerCrowdsale.sol

```
NaN function _preValidatePurchase(
NaN   address _beneficiary,
NaN   uint256 _weiAmount
NaN )
NaN   onlyIfWhitelisted(_beneficiary)
NaN   internal
NaN {
NaN   super._preValidatePurchase(_beneficiary, _weiAmount);
NaN }
```

The code meets the specification

Detail for Request 85: onlyIfWhitelisted__preValidatePurchase

08, Sep 2018

Posted by CTK report generator

1.9ms

Line 787-789 in File CelerCrowdsale.sol

```
NaN /*@CTK "onlyIfWhitelisted"
NaN   @post !this.whitelisted[_operator] -> __reverted
NaN */
```

Line 1632-1640 in File CelerCrowdsale.sol

```
NaN function _preValidatePurchase(
NaN   address _beneficiary,
NaN   uint256 _weiAmount
NaN )
NaN   onlyIfWhitelisted(_beneficiary)
NaN   internal
NaN {
NaN   super._preValidatePurchase(_beneficiary, _weiAmount);
NaN }
```

The code meets the specification

Detail for Request 86: MaxGasPriceCrowdsale_constructor

08, Sep 2018

Posted by CTK report generator

2.4ms

Line 1655-1659 in File CelerCrowdsale.sol

```
NaN /*@CTK MaxGasPriceCrowdsale_constructor
NaN   @tag assume_completion
NaN   @post _maxGasPrice > 0
NaN   @post __post.maxGasPrice == _maxGasPrice
NaN */
```

Line 1660-1663 in File CelerCrowdsale.sol

```
NaN constructor(uint256 _maxGasPrice) public {
NaN   require(_maxGasPrice > 0);
NaN   maxGasPrice = _maxGasPrice;
NaN }
```

The code meets the specification

Detail for Request 87: getUserContribution

08, Sep 2018

0.3ms

Posted by CTK report generator

Line 1699-1701 in File CelerCrowdsale.sol

```
NaN /*@CTK getUserContribution
NaN   @post __return == contributions[_beneficiary]
NaN */
```

Line 1702-1706 in File CelerCrowdsale.sol

```
NaN function getUserContribution(address _beneficiary)
NaN   public view returns (uint256)
NaN {
NaN   return contributions[_beneficiary];
NaN }
```

The code meets the specification

Detail for Request 88: MinCapCrowdsale_constructor

08, Sep 2018

2.5ms

Posted by CTK report generator

Line 1743-1746 in File CelerCrowdsale.sol

```
NaN /*@CTK MinCapCrowdsale_constructor
NaN   @tag assume_completion
NaN   @post __post.minCap == _minCap
NaN */
```

Line 1747-1750 in File CelerCrowdsale.sol

```
NaN constructor(uint256 _minCap) public {
NaN   require(_minCap > 0);
NaN   minCap = _minCap;
NaN }
```

The code meets the specification

Detail for Request 89: MinCapCrowdsale_preValidatePurchase

08, Sep 2018

Posted by CTK report generator

41.6ms

Line 1757-1761 in File CelerCrowdsale.sol

```
NaN /*@CTK MinCapCrowdsale_preValidatePurchase
NaN   @tag assume_completion
NaN   // @post __post.contributions[_beneficiary] == contributions[_beneficiary] + _wei
NaN   @post __post.contributions[_beneficiary] + _weiAmount >= minCap
NaN */
```

Line 1762-1770 in File CelerCrowdsale.sol

```
NaN function _preValidatePurchase(
NaN   address _beneficiary,
NaN   uint256 _weiAmount
NaN )
NaN   internal
NaN {
NaN   super._preValidatePurchase(_beneficiary, _weiAmount);
NaN   require(contributions[_beneficiary].add(_weiAmount) >= minCap);
NaN }
```

The code meets the specification

Detail for Request 90: constructor

08, Sep 2018

Posted by CTK report generator

10.6ms

Line 1803-1809 in File CelerCrowdsale.sol

```
NaN /*@CTK "constructor"
NaN   @tag assume_completion
NaN   @post _openingTime >= block.timestamp
NaN   @post _closingTime >= _openingTime
NaN   @post __post.openingTime == _openingTime
NaN   @post __post.closingTime == _closingTime
NaN */
```

Line 1810-1817 in File CelerCrowdsale.sol

```
NaN constructor(uint256 _openingTime, uint256 _closingTime) public {
NaN   // solium-disable-next-line security/no-block-members
NaN   require(_openingTime >= block.timestamp);
NaN   require(_closingTime >= _openingTime);
NaN
NaN   openingTime = _openingTime;
NaN   closingTime = _closingTime;
NaN }
```

The code meets the specification

Detail for Request 91: hasClosed

08, Sep 2018

Posted by CTK report generator

0.3ms

Line 1823-1825 in File CelerCrowdsale.sol

```
NaN /*@CTK hasClosed
NaN   @post __return == block.timestamp > closingTime
NaN  */
```

Line 1826-1829 in File CelerCrowdsale.sol

```
NaN function hasClosed() public view returns (bool) {
NaN   // solium-disable-next-line security/no-block-members
NaN   return block.timestamp > closingTime;
NaN }
```

The code meets the specification

Detail for Request 92: onlyWhileOpen__preValidatePurchase

08, Sep 2018

Posted by CTK report generator

2.7ms

Line 1787-1791 in File CelerCrowdsale.sol

```
NaN /*@CTK onlyWhileOpen
NaN   @tag assume_completion
NaN   @post block.timestamp >= openingTime
NaN   @post block.timestamp <= closingTime
NaN  */
```

Line 1836-1844 in File CelerCrowdsale.sol

```
NaN function _preValidatePurchase(
NaN   address _beneficiary,
NaN   uint256 _weiAmount
NaN )
NaN   internal
NaN   onlyWhileOpen
NaN {
NaN   super._preValidatePurchase(_beneficiary, _weiAmount);
NaN }
```

The code meets the specification

Detail for Request 93: StagedMaxCapCrowdsale_constructor

08, Sep 2018

Posted by CTK report generator

15.3ms

Line 1863-1869 in File CelerCrowdsale.sol

```
NaN /*@CTK StagedMaxCapCrowdsale_constructor
NaN   @tag assume_completion
NaN   @post _initialMaxCap > 0
NaN   @post _stageDuration > 0
NaN   @post __post.initialMaxCap == _initialMaxCap
NaN   @post __post.stageDuration == _stageDuration
NaN */
```

Line 1870-1876 in File CelerCrowdsale.sol

```
NaN constructor(uint256 _initialMaxCap, uint256 _stageDuration) public {
NaN   require(_initialMaxCap > 0);
NaN   require(_stageDuration > 0);
NaN
NaN   initialMaxCap = _initialMaxCap;
NaN   stageDuration = _stageDuration;
NaN }
```

The code meets the specification

Detail for Request 94: getCurrentStageIndex

08, Sep 2018

Posted by CTK report generator

471.7ms

Line 1884-1887 in File CelerCrowdsale.sol

```
NaN /*@CTK getCurrentStageIndex
NaN   @tag assume_completion
NaN   @post __return == (now - openingTime) / stageDuration + 1
NaN  */
```

Line 1888-1894 in File CelerCrowdsale.sol

```
NaN function getCurrentStageIndex() onlyWhileOpen public view returns (uint256) {
NaN   uint256 timeAfterOpening = now.sub(openingTime);
```

```
NaN   // division always truncates: http://solidity.readthedocs.io/en/v0.4.24/types.htm
NaN   uint256 tmp = timeAfterOpening.div(stageDuration);
NaN   uint256 index = tmp.add(1);
NaN   return index;
NaN }
```

The code meets the specification

Detail for Request 95: onlyWhileOpen_getCurrentStageIndex

08, Sep 2018

Posted by CTK report generator

2.6ms

Line 1787-1791 in File CelerCrowdsale.sol

```
NaN /*@CTK onlyWhileOpen
NaN   @tag assume_completion
NaN   @post block.timestamp >= openingTime
NaN   @post block.timestamp <= closingTime
NaN  */
```

Line 1888-1894 in File CelerCrowdsale.sol

```
NaN function getCurrentStageIndex() onlyWhileOpen public view returns (uint256) {
NaN   uint256 timeAfterOpening = now.sub(openingTime);
NaN   // division always truncates: http://solidity.readthedocs.io/en/v0.4.24/types.htm
NaN   uint256 tmp = timeAfterOpening.div(stageDuration);
NaN   uint256 index = tmp.add(1);
NaN   return index;
NaN }
```

The code meets the specification

Detail for Request 96: getCurrentMapCap

08, Sep 2018

Posted by CTK report generator

3983.7ms

Line 1900-1903 in File CelerCrowdsale.sol

```
NaN /*@CTK getCurrentMapCap
NaN   @tag assume_completion
NaN   @post __return == initialMaxCap * 2 ** ((now - openingTime) / stageDuration)
NaN  */
```

Line 1904-1909 in File CelerCrowdsale.sol

```
NaN function getCurrentMaxCap() public view returns (uint256) {
NaN   uint256 index = getCurrentStageIndex();
NaN   uint256 times = 2 ** (index.sub(1));
NaN   uint256 currentMaxCap = initialMaxCap.mul(times);
NaN   return currentMaxCap;
NaN }
```

The code meets the specification

Detail for Request 97: StagedMaxCapCrowdsale_preValidatePurchase

08, Sep 2018

Posted by CTK report generator

8196.2ms

Line 1916-1920 in File CelerCrowdsale.sol

```
NaN /*@CTK StagedMaxCapCrowdsale_preValidatePurchase
NaN   @tag assume_completion
NaN   @post contributions[_beneficiary] + _weiAmount <=
NaN     initialMaxCap * 2 ** ((now - openingTime) / stageDuration)
NaN  */
```

Line 1921-1930 in File CelerCrowdsale.sol

```
NaN function _preValidatePurchase(
NaN   address _beneficiary,
NaN   uint256 _weiAmount
NaN )
NaN   internal
NaN {
NaN   super._preValidatePurchase(_beneficiary, _weiAmount);
NaN   uint256 currentMaxCap = getCurrentMaxCap();
NaN   require(contributions[_beneficiary].add(_weiAmount) <= currentMaxCap);
NaN }
```

The code meets the specification

Detail for Request 98: Validate _preValidatePurchase non reverted case

08, Sep 2018

Posted by CTK report generator

4.1ms

Line 1943-1948 in File CelerCrowdsale.sol

```
NaN /*@CTK "Validate _preValidatePurchase non reverted case"
NaN   @pre this.paused == false
NaN   @pre _beneficiary != address(0)
NaN   @pre _weiAmount != 0
NaN   @post !__reverted
NaN  */
```

Line 1957-1965 in File CelerCrowdsale.sol

```
NaN function _preValidatePurchase(
NaN   address _beneficiary,
NaN   uint256 _weiAmount
NaN )
NaN   whenNotPaused
NaN   internal
NaN {
NaN   super._preValidatePurchase(_beneficiary, _weiAmount);
NaN }
```

The code meets the specification

Detail for Request 99: Validate _preValidatePurchase reverted on _beneficiary

08, Sep 2018

Posted by CTK report generator

3.5ms

Line 1949-1952 in File CelerCrowdsale.sol

```
NaN /*@CTK "Validate _preValidatePurchase reverted on _beneficiary"
NaN   @pre _beneficiary == address(0)
NaN   @post __reverted
NaN  */
```

Line 1957-1965 in File CelerCrowdsale.sol

```
NaN function _preValidatePurchase(
NaN   address _beneficiary,
NaN   uint256 _weiAmount
NaN )
NaN   whenNotPaused
NaN   internal
NaN {
NaN   super._preValidatePurchase(_beneficiary, _weiAmount);
NaN }
```

The code meets the specification

Detail for Request 100: Validate _preValidatePurchase reverted on _weiAmount

08, Sep 2018

Posted by CTK report generator

4.1ms

Line 1953-1956 in File CelerCrowdsale.sol

```
NaN /*@CTK "Validate _preValidatePurchase reverted on _weiAmount"
NaN   @pre _weiAmount == 0
NaN   @post __reverted
NaN  */
```

Line 1957-1965 in File CelerCrowdsale.sol

```
NaN function _preValidatePurchase(
NaN   address _beneficiary,
NaN   uint256 _weiAmount
NaN )
NaN   whenNotPaused
NaN   internal
NaN {
NaN   super._preValidatePurchase(_beneficiary, _weiAmount);
NaN }
```

The code meets the specification

Detail for Request 101: whenNotPaused__preValidatePurchase

08, Sep 2018

Posted by CTK report generator

2ms

Line 556-558 in File CelerCrowdsale.sol

```
NaN /*@CTK "whenNotPaused"
NaN   @post this.paused -> __reverted
NaN */
```

Line 1957-1965 in File CelerCrowdsale.sol

```
NaN function _preValidatePurchase(
NaN   address _beneficiary,
NaN   uint256 _weiAmount
NaN )
NaN   whenNotPaused
NaN   internal
NaN {
NaN   super._preValidatePurchase(_beneficiary, _weiAmount);
NaN }
```

The code meets the specification

Detail for Request 102: setRate

08, Sep 2018

17.3ms

Posted by CTK report generator

Line 2039-2045 in File CelerCrowdsale.sol

```
NaN /*@CTK setRate
NaN   @tag assume_completion
NaN   @post owner == msg.sender
NaN   @post _rate > 0
NaN   @post now < openingTime
NaN   @post __post.rate == _rate
NaN */
```

Line 2046-2051 in File CelerCrowdsale.sol

```
NaN function setRate(uint256 _rate) external onlyOwner {
NaN   require(_rate > 0);
NaN   require(now < openingTime);
NaN
NaN   rate = _rate;
NaN }
```

The code meets the specification

Detail for Request 103: onlyOwner_setRate

08, Sep 2018

2.2ms

Posted by CTK report generator

Line 487-489 in File CelerCrowdsale.sol

```
NaN /*@CTK "onlyOwner"
NaN   @post msg.sender != owner -> __reverted
NaN */
```

Line 2046-2051 in File CelerCrowdsale.sol

```
NaN function setRate(uint256 _rate) external onlyOwner {
NaN   require(_rate > 0);
NaN   require(now < openingTime);
NaN
NaN   rate = _rate;
NaN }
```

The code meets the specification

Detail for Request 104: setInitialMaxCap

08, Sep 2018

Posted by CTK report generator

16.3ms

Line 2058-2064 in File CelerCrowdsale.sol

```
NaN /*@CTK setInitialMaxCap
NaN   @tag assume_completion
NaN   @post _initialMaxCap > 0
NaN   @post now < openingTime
NaN   @post __post.initialMaxCap == _initialMaxCap
NaN   @post owner == msg.sender
NaN  */
```

Line 2065-2070 in File CelerCrowdsale.sol

```
NaN function setInitialMaxCap(uint256 _initialMaxCap) external onlyOwner {
NaN   require(_initialMaxCap > 0);
NaN   require(now < openingTime);
NaN
NaN   initialMaxCap = _initialMaxCap;
NaN }
```

The code meets the specification

Detail for Request 105: onlyOwner_setInitialMaxCap

08, Sep 2018

Posted by CTK report generator

1.7ms

Line 487-489 in File CelerCrowdsale.sol

```
NaN /*@CTK "onlyOwner"
NaN   @post msg.sender != owner -> __reverted
NaN */
```

Line 2065-2070 in File CelerCrowdsale.sol

```
NaN function setInitialMaxCap(uint256 _initialMaxCap) external onlyOwner {
NaN   require(_initialMaxCap > 0);
NaN   require(now < openingTime);
NaN
```

```
NaN    initialMaxCap = _initialMaxCap;
NaN  }
```

The code meets the specification