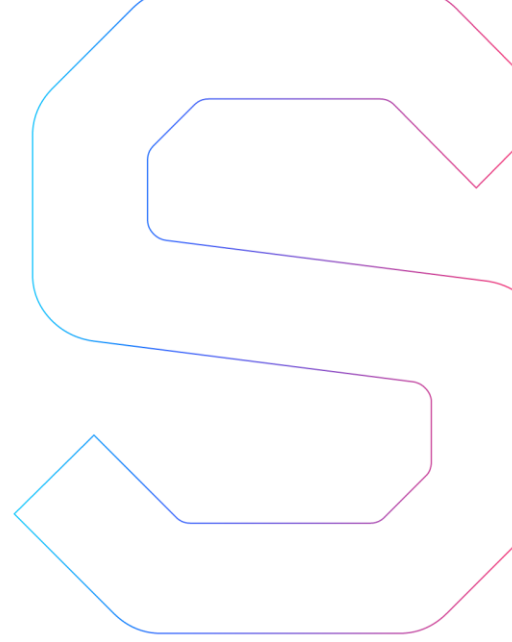


# SmartDec



## OrbsToken Smart Contracts Security Analysis

This report is public.

Published: May 29, 2018



Abstract.....	2
Disclaimer .....	2
Summary .....	2
General recommendations .....	2
Procedure .....	3
Checked vulnerabilities .....	4
Project overview.....	5
Project description .....	5
Project architecture.....	5
Code logic .....	6
Automated analysis.....	9
Manual analysis .....	11
Critical issues .....	11
Medium severity issues .....	11
Low severity issues .....	11
ERC20 approve issue .....	11
Redundant code.....	12
Code style.....	14
Misleading comments .....	14
Length of the list.....	14
No deployment script .....	15
Appendix.....	17
Compilation output.....	17
Code coverage .....	19
Tests output.....	19
Token project .....	19
Subscription project .....	36
Solhint output .....	65
Solium output .....	66

# Abstract

In this report, we consider the security of the [OrbsToken](#) project. Our task is to find and describe security issues in the smart contracts of the platform.

## Disclaimer

The audit does not give any warranties on the security of the code. One audit can not be considered enough. We always recommend proceeding with several independent audits and a public bug bounty program to ensure the security of smart contracts. Besides, security audit is not an investment advice.

## Summary

In this report we have considered the security of [OrbsToken](#) smart contracts. We performed our audit according to the [procedure](#) described below.

The audit has shown high code quality. Also, the audit has shown neither critical issues nor medium severity issues. However, a number of low severity issues were found. We recommend addressing them.

## General recommendations

The contracts code does not contain issues that endanger project security.

However, if the developer decides to improve the code, we recommend addressing low severity issues. Firstly, we recommend instructing users about [ERC20 approve issue](#). Secondly, we recommend keeping track of the [length of the federationMembers list](#) to prevent possible problems with the gas limitation when it increases. Also, we recommend removing [redundant code](#) and rewriting [several comments](#) more transparently.

However, mentioned above are minor issues. They do not influence code operation.

The text below is for technical use; it details the statements made in Summary and General recommendations.

# Procedure

In our audit, we consider the following crucial features of the smart contract code:

1. Whether the code is secure.
2. Whether the code corresponds to the documentation (including whitepaper).
3. Whether the code meets best practices in efficient use of gas, code readability, etc.

We perform our audit according to the following procedure:

- automated analysis
  - we scan project's smart contracts with our own Solidity static code analyzer [SmartCheck](#)
  - we scan project's smart contracts with several publicly available automated Solidity analysis tools such as [Remix](#), [Oyente](#), and [Solhint](#)
  - we manually verify (reject or confirm) all the issues found by tools
- manual audit
  - we manually analyze smart contracts for security vulnerabilities
  - we check smart contracts logic
  - we check ERC20 compliance
  - we run tests and check code coverage
- report
  - we reflect all the gathered information in the report

# Checked vulnerabilities

We have scanned OrbsToken smart contracts for commonly known and more specific vulnerabilities. Here are some of the commonly known vulnerabilities that we considered (the full list includes them but is not limited to them):

- [Reentrancy](#)
- [Timestamp Dependence](#)
- [Gas Limit and Loops](#)
- [DoS with \(Unexpected\) Throw](#)
- [DoS with \(Unexpected\) revert](#)
- [DoS with Block Gas Limit](#)
- [Transaction-Ordering Dependence](#)
- [Use of tx.origin](#)
- [Exception disorder](#)
- [Gasless send](#)
- [Balance equality](#)
- [Byte array](#)
- [Transfer forwards all gas](#)
- [ERC20 API violation](#)
- [Malicious libraries](#)
- [Compiler version not fixed](#)
- [Redundant fallback function](#)
- [Send instead of transfer](#)
- [Style guide violation](#)
- [Unchecked external call](#)
- [Unchecked math](#)
- [Unsafe type inference](#)
- [Implicit visibility level](#)
- [Address hardcoded](#)
- [Using delete for arrays](#)
- [Integer overflow/underflow](#)
- [Locked money](#)
- [Private modifier](#)
- [Revert/require functions](#)
- [Using var](#)
- [Visibility](#)
- [Using blockhash](#)
- [Using SHA3](#)
- [Using suicide](#)
- [Using throw](#)
- [Using inline assembly](#)

# Project overview

## Project description

In our analysis we consider smart contracts code (two repos, [token](#) and [subscription](#), versions on commits 044e1e4 and 16fb0a7 respectively).

## Project architecture

For the audit, we have been provided with the two truffle projects.

- Both projects successfully compile with `truffle compile` command (with some warnings, see [Compilation output](#) in [Appendix](#))
- The token project successfully passes all the tests (`truffle test` command, see [Tests output](#) in [Appendix](#)) with 100% code coverage
- The subscription project successfully passes all the tests (`truffle test` command, see [Tests output](#) in [Appendix](#)) with 100% code coverage

Projects are npm and yarn packages. The production deployment script was not provided for the audit. Both projects have a configuration file for Solhint linter (`.solhint.json`) and a configuration file for tests code coverage (`.solcover.js`).

The audited files from token and subscription projects are the following:

- OrbsToken.sol
- SubscriptionBilling.sol

All the other files are out of the scope of the audit.

These files contain contracts of the same name:

- OrbsToken (inherits HasNoTokens, HasNoContracts, and StandardToken contracts from [OpenZeppelin](#) library version 1.9.0)
- SubscriptionBilling (inherits HasNoContracts contract and uses SafeMath library from [OpenZeppelin](#) library version 1.9.0)

The total volume of audited files is 326 lines of Solidity code.

## Code logic

**OrbsToken** is ERC20 compatible (compatibility has been checked during the audit) token contract with the following parameters:

- token name: “Orbs”
- token symbol: “ORBS”
- token decimals: 18
- total supply: 10 billion tokens

Besides, some additional functionality was implemented:

1. The contract inherits Ownable contract. It means that the contract has an owner (firstly, it is the address the contract was deployed from).
2. The contract inherits CanReclaimToken contract. It means that in case any ERC20 compatible tokens are mistakenly sent to the contract address, the owner of the contract can transfer all the tokens to their address.
3. The contract inherits HasNoContracts contract. It means that in case ownership of any contract is transferred to the token address, the owner of the token contract can transfer ownership to themselves.
4. The contract inherits HasNoTokens contract. It means that it cannot receive any ERC223 compatible tokens.
5. The contract includes `increaseApproval` and `decreaseApproval` functions. It makes possible the `allowance` mapping changing without using `approve` function that has a vulnerability (see [ERC20 approve issue](#)).

All the tokens (total supply) are sent to the `_distributor` address passed as constructor parameter, when the OrbsToken contract is deployed.

**SubscriptionBilling** is a contract that manages subscriptions. It allows to pay for a subscription and if it does not exist yet – to create a new one and then pay for it.

List of publicly available functions with descriptions by SmartDec team:

```
constructor(OrbsToken _orbs, address[] _federationMembers,
            uint256 _minimalMonthlySubscription) public
```

Parameters requirements:

- `_orbs` address is a non-zero value
- `_federationMembers` list is valid
  - the length of the list is not null and less than `MAX_FEDERATION_MEMBERS`
  - there are no duplicates throughout the list
- `_minimalMonthlySubscription` is not null

Logic:

- contract constructor
- function initializes several variables
  - address of OrbsToken contract as `_orbs`
  - list of beneficiaries as `_federationMembers`
  - minimal fee for subscription as `_minimalMonthlySubscription`

```
function getSubscriptionData(bytes32 _id) public view returns  
(bytes32 id, string profile, uint256 startTime, uint256  
tokens)
```

Call restrictions:

- everyone can make a call

Parameters requirements:

- \_id array is not empty

Logic:

- shows id, profile, startTime, tokens for current month subscription of given id

```
function getSubscriptionDataByTime(bytes32 _id, uint16 _year,  
uint8 _month) public view returns (bytes32 id, string profile,  
uint256 startTime, uint256 tokens)
```

Call restrictions:

- everyone can make a call

Parameters requirements:

- \_id array is not empty

Logic:

- shows id, profile, startTime, tokens for specified month subscription of specified id

```
function distributeFees() public
```

Call restrictions:

- everyone can make a call

Logic:

- distributes the current month fees to the federationMembers addresses
- fees are computed in the following way
  - all the received tokens are divided between federationMembers equally
  - the remainder is sent to the first member

```
function distributeFees(uint16 _year, uint8 _month) public
```

Call restrictions:

- everyone can make a call

Parameters requirements:

- \_year and \_month must not denote the future date

Logic:

- distributes the specified month fees to the federationMembers addresses
- fees are computed in the following way
  - all the received tokens are divided between federationMembers equally
  - the remainder is sent to the first member

```
function subscribeForCurrentMonth(bytes32 _id, string  
_profile, uint256 _value) public
```

Call restrictions:

- everyone can make a call

Parameters requirements:



- `_id` array is not empty
- `_profile` string is not empty
- `_value` should be greater than 0
- user must approve enough tokens (greater than or equal to `_value`) before calling this function

Logic:

- works with subscriptions of the current month
- transfers `_value` tokens to the contract address
- in case subscription does not exist it creates a new one
  - `id` equalling `_id`
  - `profile` equalling `_profile`
  - `startTime` equalling the current time
- computes `subscription.tokens` – the overall payment for the current month subscription with the specified id
  - the overall payment must be greater than `minimalMonthlySubscription`
- computes the overall payment for all the current month subscriptions

```
function subscribeForNextMonth(bytes32 _id, string _profile,
uint256 _value) public
```

Call restrictions:

- everyone can make a call

Parameters requirements:

- `_id` array is not empty
- `_profile` string is not empty
- `_value` should be greater than 0
- user must approve enough tokens (greater than or equal to `_value`) before calling this function

Logic:

- works with subscriptions of the next month
- transfers `_value` tokens to the contract address
- in case subscription does not exist it creates a new one
  - `id` equalling `_id`
  - `profile` equalling `_profile`
  - `startTime` equalling the beginning of the next month
- computes `subscription.tokens` – the overall payment for the next month subscription with the specified id
  - the overall payment must be greater than `minimalMonthlySubscription`
- computes the overall payment for all the next month subscriptions

# Automated analysis

We used several publicly available automated Solidity analysis tools. Here are the combined results of SmartCheck, Solhint, and Remix. Oyente has found no issues.

All the issues found by tools were manually checked (rejected or confirmed).

**False positives** are constructions that were discovered by the tools as vulnerabilities but do not consist a security threat.

**True positives** are constructions that were discovered by the tools as vulnerabilities and can actually be exploited by attackers or lead to incorrect contracts operation.

Cases when these issues lead to actual bugs or vulnerabilities are described in the next section.

Tool	Rule	False positives	True positives
Remix	Gas requirement of function high	0	24
	Is constant but potentially should not be.	0	4
	Potential Violation of Checks-Effects-Interaction pattern	0	2
	Potentially should be constant but is not.	0	2
	use of "now"	0	3
	Variables have very similar names	0	3
	Total Remix	0	38
SmartCheck	Address Hardcoded	0	1
	Dos With Revert	0	3
	Gas Limit And Loops	0	3

No Payable Fallback	0	3
Reentrancy External Call	0	10
Timestamp Dependence	0	1
Unchecked Math	0	3
Visibility	0	2
Total SmartCheck	0	26
<b>Total Overall</b>	<b>0</b>	<b>64</b>

# Manual analysis

The contracts were completely manually analyzed. Besides, the results of the automated analysis were manually verified. All confirmed issues are described below.

## Critical issues

Critical issues seriously endanger smart contracts security. We highly recommend fixing them.

**The audit has shown no critical issues.**

## Medium severity issues

Medium issues can influence smart contracts operation in current implementation. We highly recommend addressing them.

**The audit has shown no medium severity issues.**

## Low severity issues

Low severity issues can influence smart contracts operation in future versions of code. We recommend taking them into account.

### ERC20 approve issue

There is [ERC20 approve issue](#): changing the approved amount from a nonzero value to another nonzero value allows a double spending with a front-running attack.

We recommend instructing users to follow one of two ways:

- not to use `approve()` function directly and to use `increaseApproval()`/`decreaseApproval()` functions instead
- to change the approved amount to 0, wait for the transaction to be mined, and then to change the approved amount to the desired value

## Redundant code

The following lines include redundant code:

1. SubscriptionBilling.sol, line 184: adding `require()` is useless here as `transferFrom()` function always returns `true`. Moreover, in case user hasn't allocated enough tokens `transferFrom()` function will revert the transaction.  
Comment from the developer:  
"This is a matter of style. In our implementation, `transferFrom()` indeed can never return false, but according to the standard - it might for other tokens, so for the sake of safety - I'd prefer to have this test for future implementations and versions."
2. SubscriptionBilling.sol, line 46: the constant `EMPTY` is declared with a `public` modifier. The `public` modifier creates a getter function automatically, however in this case created getter function will always return zero (the initial value of the `EMPTY` constant). There is no sense in this getter function, that is why we recommend using `private` modifier for the constant. There are a few more variables with `public` modifier in the code, we recommend considering the need for automatic getters for these variables. In case getters are not needed, you should use `private` modifier for these variables.
3. SubscriptionBilling.sol, line 181:

```
require(_startTime >= now, "Starting time must be in the future");
```

This check is redundant in the current implementation. Because in both places where this function is called, the passed `_startTime` variable is greater or equal to `now`.

Comment from the developer:

"This test is required in order to reach full test code coverage. In addition, a future factoring can make this method `public/external`, in which case this test should be there."

4. DateTime.sol: `getHour`, `getMinute`, `getSecond`, `getWeekday`, `getDay` functions are not used anywhere, so they should be removed. Also, `getMonth` function is used only once, in addition, it contains the calculation of seconds, minutes, hours, days and weeks, which in our case is unnecessary and only increases the cost of this operation. We recommend removing redundant functionality of DateTime library and optimizing `getMonth` function.  
Comment from the developer:  
"This is a library that can and should be used elsewhere. It's an incredibly bad practice and a premature optimization to start messing with it for the sake of deploy gas cost optimizations."
5. SubscriptionBilling.sol, line 124: using `div` function from the SafeMath library is unnecessary:

```
uint256 fee = monthlySubscription  
.totalTokens.div(federationMembers.length);
```

Because contract constructor requires `federationMembers.length` to be greater than zero. Also, solidity automatically throws when dividing by 0.

6. SubscriptionBilling.sol, line 133: using add function from the SafeMath library is unnecessary in this case:

```
memberFee = memberFee.add(monthlySubscription.totalTokens %  
federationMembers.length);
```

Because the first term is the result of division, line 124:

```
monthlySubscription.totalTokens.div(federationMembers.length);
```

and the second term is remainder of this division. Finally, the sum of the result of the division and the remainder of this division cannot be greater than the original number (monthlySubscription.totalTokens), hence the overflow is impossible here.

7. SubscriptionBilling.sol, line 136: using sub function from the SafeMath library is unnecessary in this case:

```
monthlySubscription.totalTokens =  
monthlySubscription.totalTokens.sub(memberFee);
```

Because memberFee variable is equal to

```
monthlySubscription.totalTokens.div(federationMembers.length);
```

and subtraction repeats exactly federationMembers.length times. Also, at first iteration the remainder of division is subtracted. Thus, after this loop monthlySubscription.totalTokens will be equal to 0. Thereby, underflow is impossible in this case.

8. SubscriptionBilling.sol, line 202,208: using add function from the SafeMath library is unnecessary in these cases:

```
subscription.tokens = subscription.tokens.add(_value);  
  
monthlySubscription.totalTokens =  
monthlySubscription.totalTokens.add(_value);
```

Because the \_value variable is the real number of tokens received from the user, line 184:

```
require(orbs.transferFrom(msg.sender, address(this),  
_value), "Insufficient allowance!");
```

these sums cannot be more than the total number of tokens (overflow is impossible). In general, we recommend adding explicit checks rather than using SafeMath library.

Comment from the developer on paragraphs 5-8:

"This is another style issue and precaution. There shouldn't be any arithmetic operation that isn't using SafeMath, regardless if we can prove that no overflow/underflow is possible."

We highly recommend removing redundant code for greater readability and transparency of code and less cost of deployment and execution.

## Code style

We found in the code, that the contract variable is used as the address variable, SubscriptionBilling.sol, line 57:

```
require(_orbs != address(0), "Address must not be 0!");
```

We draw attention to the fact that in recent versions of the compiler, this practice is considered to be incorrect. Instead, we recommend using

```
address(_orbs) != address(0)
```

*The issue has been fixed by the developer and is not present on commit a0845e0d.*

## Misleading comments

Controversial comments were found in several places of the code:

1. SubscriptionBilling.sol, line 171: the comment says:

```
// This method needs to be called by an ERC223 transfer  
with a valid amount of Orbs tokens.
```

However, the contract cannot receive ERC223 tokens. To do this, tokenFallback function is required. However, it is not implemented in the considered version of the contract.

*The issue has been fixed by the developer and is not present on commit a380e8e6.*

2. SubscriptionBilling.sol, line 207:

```
// Update this month's total subscription allocations.
```

At this point, we update the totalTokens value for the selected month, and not for the current month as one might think after reading the comments.

*The issue has been fixed by the developer and is not present on commit 4852c912.*

## Length of the list

We noticed, that the federationMembers list can only be specified in the constructor of the contract, where it is checked by isFederationMembersListValid function. The longer the list length is, the more gas the function execution requires as it includes a loop with \_federationMembers.length-1 iterations, line 237:

```
for (uint i = 0; i < _federationMembers.length - 1; ++i)
```

Consequently, it is impossible to make this list large enough due to the gas restrictions. In this implementation of the contract, the MAX\_FEDERATION\_MEMBERS constant is one hundred and there is no gas issue. However, if developers want to make the federationMembers list larger, or if developers need to update it in the existing contract, it

is probably worth implementing a service function, that will update the list. Thus, it will be possible to fill the list partially, avoiding gas limitations problems.

There is also a loop that iterates through this array at SubscriptionBilling.sol, line 127:

```
for (uint i = 0; i < federationMembers.length; ++i)
```

here you can implement withdraw pattern instead of distribute pattern, if you want to increase the length of the `federationMembers` list.

Moreover, `isFederationMembersListValid` function checks for duplicates through a square of complexity. We recommend transferring the sorted array to the constructor, and check the sorting and duplicates for linear time.

Comment from the developer:

"Length of the list - this is exactly why MAX\_FEDERATION\_MEMBERS is for and why the tests make sure to test this scenario as well. If someone changes MAX\_FEDERATION\_MEMBERS to 100000 tomorrow - the tests will just fail and it'd be evident that pagination is needed."

## No deployment script

We have not received a production deployment script for our audit. That is why we recommend users check critical parameters such as token address, after SubscriptionBilling contract is deployed.

Comment from the developer:

"The deployment will be done manually intentionally."



This analysis was performed by [SmartDec](#).

Yaroslav Alexandrov, Head of Development Department  
Evgeniy Marchenko, Lead Developer  
Ivan Ivanitskiy, Chief Analytics Officer  
Artyom Orlov, Business Development Manager  
Igor Sobolev, Analyst  
Alexander Drygin, Analyst

Sergey Pavlin, Chief Operating Officer

A handwritten signature in black ink, appearing to read 'Pavlin', with a stylized flourish at the end.

May 29, 2018

# Appendix

## Compilation output

```
//Token project
Compiling ./contracts/Migrations.sol...
Compiling ./contracts/MultiSigWallet.sol...
Compiling ./contracts/OrbsToken.sol...
Compiling ./test/helpers/MultiSigWalletMock.sol...
Compiling ./test/helpers/TestERC20Token.sol...
Compiling zeppelin-solidity/contracts/math/SafeMath.sol...
Compiling zeppelin-
solidity/contracts/ownership/CanReclaimToken.sol...
Compiling zeppelin-
solidity/contracts/ownership/HasNoContracts.sol...
Compiling zeppelin-
solidity/contracts/ownership/HasNoTokens.sol...
Compiling zeppelin-solidity/contracts/ownership/Ownable.sol...
Compiling zeppelin-
solidity/contracts/token/ERC20/BasicToken.sol...
Compiling zeppelin-solidity/contracts/token/ERC20/ERC20.sol...
Compiling zeppelin-
solidity/contracts/token/ERC20/ERC20Basic.sol...
Compiling zeppelin-
solidity/contracts/token/ERC20/SafeERC20.sol...
Compiling zeppelin-
solidity/contracts/token/ERC20/StandardToken.sol...

Compilation warnings encountered:

zeppelin-solidity/contracts/ownership/Ownable.sol:20:3:
Warning: Defining constructors as functions with the same name
as the contract is deprecated. Use "constructor(...) { ... }"
instead.
    function Ownable() public {
      ^ (Relevant source part starts here and spans across
multiple lines).
,zeppelin-solidity/contracts/ownership/HasNoTokens.sol:21:3:
Warning: Function state mutability can be restricted to pure
    function tokenFallback(address from_, uint256 value_, bytes
data_) external {
```

^ (Relevant source part starts here and spans across multiple lines).

```
//Subscription project
Starting our own ganache instance
Compiling ./contracts/DateTime.sol...
Compiling ./contracts/Migrations.sol...
Compiling ./contracts/OrbsToken.sol...
Compiling ./contracts/SubscriptionBilling.sol...
Compiling ./test/DateTimeWrapper.sol...
Compiling ./test/OrbsTokenMock.sol...
Compiling ./test/SubscriptionBillingMock.sol...
Compiling zeppelin-solidity/contracts/math/SafeMath.sol...
Compiling zeppelin-
solidity/contracts/ownership/CanReclaimToken.sol...
Compiling zeppelin-
solidity/contracts/ownership/HasNoContracts.sol...
Compiling zeppelin-
solidity/contracts/ownership/HasNoTokens.sol...
Compiling zeppelin-solidity/contracts/ownership/Ownable.sol...
Compiling zeppelin-
solidity/contracts/token/ERC20/BasicToken.sol...
Compiling zeppelin-solidity/contracts/token/ERC20/ERC20.sol...
Compiling zeppelin-
solidity/contracts/token/ERC20/ERC20Basic.sol...
Compiling zeppelin-
solidity/contracts/token/ERC20/SafeERC20.sol...
Compiling zeppelin-
solidity/contracts/token/ERC20/StandardToken.sol...
```

Compilation warnings encountered:

```
zeppelin-solidity/contracts/ownership/Ownable.sol:20:3:
Warning: Defining constructors as functions with the same name
as the contract is deprecated. Use "constructor(...) { ... }"
instead.
```

```
    function Ownable() public {
      ^ (Relevant source part starts here and spans across
multiple lines).
,zeppelin-solidity/contracts/ownership/HasNoTokens.sol:21:3:
Warning: Function state mutability can be restricted to pure
    function tokenFallback(address from_, uint256 value_, bytes
data_) external {
      ^ (Relevant source part starts here and spans across
multiple lines).
```

## Code coverage

File	% Stmts	% Branch	% Funcs	% Lines	Uncovered Lines
OrbsToken.sol	100	100	100	100	
SubscriptionBilling.sol	100	100	100	100	
All files	100	100	100	100	

## Tests output

### Token project

```
Starting our own ganache instance
Compiling ./contracts/Migrations.sol...
Compiling ./contracts/MultiSigWallet.sol...
Compiling ./contracts/OrbsToken.sol...
Compiling ./test/helpers/MultiSigWalletMock.sol...
Compiling ./test/helpers/TestERC20Token.sol...
Compiling zeppelin-solidity/contracts/math/SafeMath.sol...
Compiling zeppelin-
solidity/contracts/ownership/CanReclaimToken.sol...
Compiling zeppelin-
solidity/contracts/ownership/HasNoContracts.sol...
Compiling zeppelin-
solidity/contracts/ownership/HasNoTokens.sol...
Compiling zeppelin-solidity/contracts/ownership/Ownable.sol...
Compiling zeppelin-
solidity/contracts/token/ERC20/BasicToken.sol...
Compiling zeppelin-solidity/contracts/token/ERC20/ERC20.sol...
Compiling zeppelin-
solidity/contracts/token/ERC20/ERC20Basic.sol...
Compiling zeppelin-
solidity/contracts/token/ERC20/SafeERC20.sol...
Compiling zeppelin-
solidity/contracts/token/ERC20/StandardToken.sol...
```

Compilation warnings encountered:

```
zeppelin-solidity/contracts/ownership/Ownable.sol:20:3:
Warning: Defining constructors as functions with the same name
as the contract is deprecated. Use "constructor(...) { ... }"
instead.
```

```

function Ownable() public {
  ^ (Relevant source part starts here and spans across
multiple lines).
,zeppelin-solidity/contracts/ownership/HasNoTokens.sol:21:3:
Warning: Function state mutability can be restricted to pure
  function tokenFallback(address from_, uint256 value_, bytes
data_) external {
  ^ (Relevant source part starts here and spans across
multiple lines).

[0m[0m
[0m  Contract: MultiSigWallet[0m
[0m    construction[0m
[0m      error[0m
      [32m  ✓[0m[90m should throw if created with more than 50
owners[0m
      [32m  ✓[0m[90m should throw if created without any
owners[0m
      [32m  ✓[0m[90m should throw if created without any
requirements[0m
      [32m  ✓[0m[90m should throw if created with a
requirement larger than the number of owners[0m
      [32m  ✓[0m[90m should throw if created with duplicate
owners[0m
[0m      success[0m
      [32m  ✓[0m[90m should be initialized with 0 balance[0m
      [32m  ✓[0m[90m should initialize owners[0m
      [32m  ✓[0m[90m should initialize owners's mapping[0m
      [32m  ✓[0m[90m should initialize requirement[0m
      [32m  ✓[0m[90m should initialize with empty transaction
count[0m
[0m    fallback function[0m
      [32m  ✓[0m[90m should not log empty deposits[0m
      [32m  ✓[0m[90m should receive ETH[0m
      [32m  ✓[0m[90m should receive ERC20[0m
[0m    transaction submission and confirmation[0m
[0m      with 2 owners and requirement of 1[0m
      [32m  ✓[0m[90m should only send ETH when all
confirmations were received[0m
      [32m  ✓[0m[90m should only send ERC20 when all
confirmations were received[0m
[0m      submitTransaction[0m
      [32m  ✓[0m[90m should throw an error, if sent from not

```

```

an owner[0m
    [32m ✓[0m[90m should throw an error, if sent to a 0
address[0m
[0m    confirmTransaction[0m
    [32m ✓[0m[90m should throw an error, if confirming
the same transaction after submitting it[0m
    [32m ✓[0m[90m should throw an error, if confirming a
non-existing transaction[0m
[0m    revokeConfirmation[0m
    [32m ✓[0m[90m should throw an error, if asked to
revoke an executed transaction[0m
[0m    with 2 owners and requirement of 2[0m
    [32m ✓[0m[90m should only send ETH when all
confirmations were received[0m
    [32m ✓[0m[90m should only send ERC20 when all
confirmations were received[0m
[0m    submitTransaction[0m
    [32m ✓[0m[90m should throw an error, if sent from not
an owner[0m
    [32m ✓[0m[90m should throw an error, if sent to a 0
address[0m
[0m    confirmTransaction[0m
    [32m ✓[0m[90m should throw an error, if confirming
the same transaction after submitting it[0m
    [32m ✓[0m[90m should throw an error, if sent from not
an owner[0m
    [32m ✓[0m[90m should throw an error, if confirming
the same transaction twice[0m
    [32m ✓[0m[90m should throw an error, if confirming a
non-existing transaction[0m
[0m    revokeConfirmation[0m
    [32m ✓[0m[90m should throw an error, if sent from not
an owner[0m
    [32m ✓[0m[90m should throw an error, if asked to
revoke a non-confirmed transaction[0m
    [32m ✓[0m[90m should throw an error, if asked to
revoke an executed transaction[0m
[0m    executeTransaction[0m
[0m    executed[0m
    [32m ✓[0m[90m should throw, if called again[0m
[0m    not executed[0m
[0m    not confirmed[0m
    [32m ✓[0m[90m should be executed automatically,
once final confirmation is received[0m

```

```

[32m  ✓[0m[90m should fail gracefully, if called
by a confirmed owner[0m
[32m  ✓[0m[90m should throw, if called by a non-
confirmed owner[0m
[0m          confirmed[0m
[32m  ✓[0m[90m should throw an error, if asked to
be executed by not an owner[0m
[32m  ✓[0m[90m should be executed successfully
when retrying[0m
[0m          with 3 owners and requirement of 2[0m
[32m  ✓[0m[90m should only send ETH when all
confirmations were received[0m
[32m  ✓[0m[90m should only send ERC20 when all
confirmations were received[0m
[0m          submitTransaction[0m
[32m  ✓[0m[90m should throw an error, if sent from not
an owner[0m
[32m  ✓[0m[90m should throw an error, if sent to a 0
address[0m
[0m          confirmTransaction[0m
[32m  ✓[0m[90m should throw an error, if confirming
the same transaction after submitting it[0m
[32m  ✓[0m[90m should throw an error, if sent from not
an owner[0m
[32m  ✓[0m[90m should throw an error, if confirming
the same transaction twice[0m
[32m  ✓[0m[90m should throw an error, if confirming a
non-existing transaction[0m
[0m          revokeConfirmation[0m
[32m  ✓[0m[90m should throw an error, if sent from not
an owner[0m
[32m  ✓[0m[90m should throw an error, if asked to
revoke a non-confirmed transaction[0m
[32m  ✓[0m[90m should throw an error, if asked to
revoke an executed transaction[0m
[0m          executeTransaction[0m
[0m          executed[0m
[32m  ✓[0m[90m should throw, if called again[0m
[0m          not executed[0m
[0m          not confirmed[0m
[32m  ✓[0m[90m should be executed automatically,
once final confirmation is received[0m
[32m  ✓[0m[90m should fail gracefully, if called

```

```

by a confirmed owner[0m
    [32m ✓[0m[90m should throw, if called by a non-
confirmed owner[0m
[0m        confirmed[0m
    [32m ✓[0m[90m should throw an error, if asked to
be executed by not an owner[0m
    [32m ✓[0m[90m should throw an error, if asked to
be executed by an unconfirmed owner[0m
    [32m ✓[0m[90m should be executed successfully
when retrying[0m
[0m        with 3 owners and requirement of 3[0m
    [32m ✓[0m[90m should only send ETH when all
confirmations were received[0m
    [32m ✓[0m[90m should only send ERC20 when all
confirmations were received[0m
[0m        submitTransaction[0m
    [32m ✓[0m[90m should throw an error, if sent from not
an owner[0m
    [32m ✓[0m[90m should throw an error, if sent to a 0
address[0m
[0m        confirmTransaction[0m
    [32m ✓[0m[90m should throw an error, if confirming
the same transaction after submitting it[0m
    [32m ✓[0m[90m should throw an error, if sent from not
an owner[0m
    [32m ✓[0m[90m should throw an error, if confirming
the same transaction twice[0m
    [32m ✓[0m[90m should throw an error, if confirming a
non-existing transaction[0m
[0m        revokeConfirmation[0m
    [32m ✓[0m[90m should throw an error, if sent from not
an owner[0m
    [32m ✓[0m[90m should throw an error, if asked to
revoke a non-confirmed transaction[0m
    [32m ✓[0m[90m should revoke a confirmation[0m
    [32m ✓[0m[90m should throw an error, if asked to
revoke an executed transaction[0m
[0m        executeTransaction[0m
[0m        executed[0m
    [32m ✓[0m[90m should throw, if called again[0m
[0m        not executed[0m
[0m        not confirmed[0m
    [32m ✓[0m[90m should be executed automatically,

```



```

once final confirmation is received[0m
    [32m ✓[0m[90m should fail gracefully, if called
by a confirmed owner[0m
    [32m ✓[0m[90m should throw, if called by a non-
confirmed owner[0m
[0m        confirmed[0m
    [32m ✓[0m[90m should throw an error, if asked to
be executed by not an owner[0m
    [32m ✓[0m[90m should be executed successfully
when retrying[0m
[0m        with 4 owners and requirement of 1[0m
    [32m ✓[0m[90m should only send ETH when all
confirmations were received[0m
    [32m ✓[0m[90m should only send ERC20 when all
confirmations were received[0m
[0m        submitTransaction[0m
    [32m ✓[0m[90m should throw an error, if sent from not
an owner[0m
    [32m ✓[0m[90m should throw an error, if sent to a 0
address[0m
[0m        confirmTransaction[0m
    [32m ✓[0m[90m should throw an error, if confirming
the same transaction after submitting it[0m
    [32m ✓[0m[90m should throw an error, if confirming a
non-existing transaction[0m
[0m        revokeConfirmation[0m
    [32m ✓[0m[90m should throw an error, if asked to
revoke an executed transaction[0m
[0m        with 4 owners and requirement of 2[0m
    [32m ✓[0m[90m should only send ETH when all
confirmations were received[0m
    [32m ✓[0m[90m should only send ERC20 when all
confirmations were received[0m
[0m        submitTransaction[0m
    [32m ✓[0m[90m should throw an error, if sent from not
an owner[0m
    [32m ✓[0m[90m should throw an error, if sent to a 0
address[0m
[0m        confirmTransaction[0m
    [32m ✓[0m[90m should throw an error, if confirming
the same transaction after submitting it[0m
    [32m ✓[0m[90m should throw an error, if sent from not
an owner[0m

```

```

[32m ✓[0m[90m should throw an error, if confirming
the same transaction twice[0m
[32m ✓[0m[90m should throw an error, if confirming a
non-existing transaction[0m
[0m      revokeConfirmation[0m
[32m ✓[0m[90m should throw an error, if sent from not
an owner[0m
[32m ✓[0m[90m should throw an error, if asked to
revoke a non-confirmed transaction[0m
[32m ✓[0m[90m should throw an error, if asked to
revoke an executed transaction[0m
[0m      executeTransaction[0m
[0m      executed[0m
[32m ✓[0m[90m should throw, if called again[0m
[0m      not executed[0m
[0m      not confirmed[0m
[32m ✓[0m[90m should be executed automatically,
once final confirmation is received[0m
[32m ✓[0m[90m should fail gracefully, if called
by a confirmed owner[0m
[32m ✓[0m[90m should throw, if called by a non-
confirmed owner[0m
[0m      confirmed[0m
[32m ✓[0m[90m should throw an error, if asked to
be executed by not an owner[0m
[32m ✓[0m[90m should throw an error, if asked to
be executed by an unconfirmed owner[0m
[32m ✓[0m[90m should be executed successfully
when retrying[0m
[0m      with 4 owners and requirement of 3[0m
[32m ✓[0m[90m should only send ETH when all
confirmations were received[0m
[32m ✓[0m[90m should only send ERC20 when all
confirmations were received[0m
[0m      submitTransaction[0m
[32m ✓[0m[90m should throw an error, if sent from not
an owner[0m
[32m ✓[0m[90m should throw an error, if sent to a 0
address[0m
[0m      confirmTransaction[0m
[32m ✓[0m[90m should throw an error, if confirming
the same transaction after submitting it[0m
[32m ✓[0m[90m should throw an error, if sent from not

```

```

an owner[0m
    [32m ✓[0m[90m should throw an error, if confirming
the same transaction twice[0m
    [32m ✓[0m[90m should throw an error, if confirming a
non-existing transaction[0m
[0m         revokeConfirmation[0m
    [32m ✓[0m[90m should throw an error, if sent from not
an owner[0m
    [32m ✓[0m[90m should throw an error, if asked to
revoke a non-confirmed transaction[0m
    [32m ✓[0m[90m should revoke a confirmation[0m
    [32m ✓[0m[90m should throw an error, if asked to
revoke an executed transaction[0m
[0m         executeTransaction[0m
[0m         executed[0m
    [32m ✓[0m[90m should throw, if called again[0m
[0m         not executed[0m
[0m         not confirmed[0m
    [32m ✓[0m[90m should be executed automatically,
once final confirmation is received[0m
    [32m ✓[0m[90m should fail gracefully, if called
by a confirmed owner[0m
    [32m ✓[0m[90m should throw, if called by a non-
confirmed owner[0m
[0m         confirmed[0m
    [32m ✓[0m[90m should throw an error, if asked to
be executed by not an owner[0m
    [32m ✓[0m[90m should throw an error, if asked to
be executed by an unconfirmed owner[0m
    [32m ✓[0m[90m should be executed successfully
when retrying[0m
[0m         with 4 owners and requirement of 4[0m
    [32m ✓[0m[90m should only send ETH when all
confirmations were received[0m
    [32m ✓[0m[90m should only send ERC20 when all
confirmations were received[0m
[0m         submitTransaction[0m
    [32m ✓[0m[90m should throw an error, if sent from not
an owner[0m
    [32m ✓[0m[90m should throw an error, if sent to a 0
address[0m
[0m         confirmTransaction[0m
    [32m ✓[0m[90m should throw an error, if confirming

```

```

the same transaction after submitting it[0m
    [32m ✓[0m[90m should throw an error, if sent from not
an owner[0m
    [32m ✓[0m[90m should throw an error, if confirming
the same transaction twice[0m
    [32m ✓[0m[90m should throw an error, if confirming a
non-existing transaction[0m
[0m         revokeConfirmation[0m
    [32m ✓[0m[90m should throw an error, if sent from not
an owner[0m
    [32m ✓[0m[90m should throw an error, if asked to
revoke a non-confirmed transaction[0m
    [32m ✓[0m[90m should revoke a confirmation[0m
    [32m ✓[0m[90m should throw an error, if asked to
revoke an executed transaction[0m
[0m         executeTransaction[0m
[0m         executed[0m
    [32m ✓[0m[90m should throw, if called again[0m
[0m         not executed[0m
[0m         not confirmed[0m
    [32m ✓[0m[90m should be executed automatically,
once final confirmation is received[0m
    [32m ✓[0m[90m should fail gracefully, if called
by a confirmed owner[0m
    [32m ✓[0m[90m should throw, if called by a non-
confirmed owner[0m
[0m         confirmed[0m
    [32m ✓[0m[90m should throw an error, if asked to
be executed by not an owner[0m
    [32m ✓[0m[90m should be executed successfully
when retrying[0m
[0m         with 5 owners and requirement of 3[0m
    [32m ✓[0m[90m should only send ETH when all
confirmations were received[0m
    [32m ✓[0m[90m should only send ERC20 when all
confirmations were received[0m
[0m         submitTransaction[0m
    [32m ✓[0m[90m should throw an error, if sent from not
an owner[0m
    [32m ✓[0m[90m should throw an error, if sent to a 0
address[0m
[0m         confirmTransaction[0m
    [32m ✓[0m[90m should throw an error, if confirming

```

```

the same transaction after submitting it[0m
    [32m ✓[0m[90m should throw an error, if sent from not
an owner[0m
    [32m ✓[0m[90m should throw an error, if confirming
the same transaction twice[0m
    [32m ✓[0m[90m should throw an error, if confirming a
non-existing transaction[0m
[0m         revokeConfirmation[0m
    [32m ✓[0m[90m should throw an error, if sent from not
an owner[0m
    [32m ✓[0m[90m should throw an error, if asked to
revoke a non-confirmed transaction[0m
    [32m ✓[0m[90m should revoke a confirmation[0m
    [32m ✓[0m[90m should throw an error, if asked to
revoke an executed transaction[0m
[0m         executeTransaction[0m
[0m         executed[0m
    [32m ✓[0m[90m should throw, if called again[0m
[0m         not executed[0m
[0m         not confirmed[0m
    [32m ✓[0m[90m should be executed automatically,
once final confirmation is received[0m
    [32m ✓[0m[90m should fail gracefully, if called
by a confirmed owner[0m
    [32m ✓[0m[90m should throw, if called by a non-
confirmed owner[0m
[0m         confirmed[0m
    [32m ✓[0m[90m should throw an error, if asked to
be executed by not an owner[0m
    [32m ✓[0m[90m should throw an error, if asked to
be executed by an unconfirmed owner[0m
    [32m ✓[0m[90m should be executed successfully
when retrying[0m
[0m         internal methods[0m
[0m         with 2 owners and requirement of 1[0m
[0m         addOwner[0m
    [32m ✓[0m[90m should throw an error, if called
directly[0m
    [32m ✓[0m[90m should throw an error, if called by not
an owner[0m
    [32m ✓[0m[90m should throw an error, if adding an
empty owner[0m
    [32m ✓[0m[90m should throw an error, if adding an

```

```

existing owner[0m
    [32m ✓[0m[90m should add an owner[0m
[0m    removeOwner[0m
    [32m ✓[0m[90m should throw an error, if called
directly[0m
    [32m ✓[0m[90m should throw an error, if called by not
an owner[0m
    [32m ✓[0m[90m should throw an error, if removing a
non-existing owner[0m
    [32m ✓[0m[90m should remove an owner[0m
[0m    replaceOwner[0m
    [32m ✓[0m[90m should throw an error, if called
directly[0m
    [32m ✓[0m[90m should throw an error, if called by not
an owner[0m
    [32m ✓[0m[90m should throw an error, if replacing a
non-existing owner[0m
    [32m ✓[0m[90m should replace an owner[0m
[0m    changeRequirement[0m
    [32m ✓[0m[90m should throw an error, if called
directly[0m
    [32m ✓[0m[90m should throw an error, if called by not
an owner[0m
    [32m ✓[0m[90m should increase requirement by 1[0m
[0m    with 2 owners and requirement of 2[0m
[0m    addOwner[0m
    [32m ✓[0m[90m should throw an error, if called
directly[0m
    [32m ✓[0m[90m should throw an error, if called by not
an owner[0m
    [32m ✓[0m[90m should throw an error, if adding an
empty owner[0m
    [32m ✓[0m[90m should throw an error, if adding an
existing owner[0m
    [32m ✓[0m[90m should add an owner[0m
[0m    removeOwner[0m
    [32m ✓[0m[90m should throw an error, if called
directly[0m
    [32m ✓[0m[90m should throw an error, if called by not
an owner[0m
    [32m ✓[0m[90m should throw an error, if removing a
non-existing owner[0m
    [32m ✓[0m[90m should remove an owner[0m

```

```

[0m         replaceOwner[0m
          [32m ✓[0m[90m should throw an error, if called
directly[0m
          [32m ✓[0m[90m should throw an error, if called by not
an owner[0m
          [32m ✓[0m[90m should throw an error, if replacing a
non-existing owner[0m
          [32m ✓[0m[90m should replace an owner[0m
[0m         changeRequirement[0m
          [32m ✓[0m[90m should throw an error, if called
directly[0m
          [32m ✓[0m[90m should throw an error, if called by not
an owner[0m
          [32m ✓[0m[90m should decrease requirement by 1[0m
[0m         with 3 owners and requirement of 2[0m
[0m         addOwner[0m
          [32m ✓[0m[90m should throw an error, if called
directly[0m
          [32m ✓[0m[90m should throw an error, if called by not
an owner[0m
          [32m ✓[0m[90m should throw an error, if adding an
empty owner[0m
          [32m ✓[0m[90m should throw an error, if adding an
existing owner[0m
          [32m ✓[0m[90m should add an owner[0m
[0m         removeOwner[0m
          [32m ✓[0m[90m should throw an error, if called
directly[0m
          [32m ✓[0m[90m should throw an error, if called by not
an owner[0m
          [32m ✓[0m[90m should throw an error, if removing a
non-existing owner[0m
          [32m ✓[0m[90m should remove an owner[0m
[0m         replaceOwner[0m
          [32m ✓[0m[90m should throw an error, if called
directly[0m
          [32m ✓[0m[90m should throw an error, if called by not
an owner[0m
          [32m ✓[0m[90m should throw an error, if replacing a
non-existing owner[0m
          [32m ✓[0m[90m should replace an owner[0m
[0m         changeRequirement[0m
          [32m ✓[0m[90m should throw an error, if called

```

```

directly[0m
    [32m ✓[0m[90m should throw an error, if called by not
an owner[0m
    [32m ✓[0m[90m should increase requirement by 1[0m
[0m    with 3 owners and requirement of 3[0m
[0m    addOwner[0m
    [32m ✓[0m[90m should throw an error, if called
directly[0m
    [32m ✓[0m[90m should throw an error, if called by not
an owner[0m
    [32m ✓[0m[90m should throw an error, if adding an
empty owner[0m
    [32m ✓[0m[90m should throw an error, if adding an
existing owner[0m
    [32m ✓[0m[90m should add an owner[0m
[0m    removeOwner[0m
    [32m ✓[0m[90m should throw an error, if called
directly[0m
    [32m ✓[0m[90m should throw an error, if called by not
an owner[0m
    [32m ✓[0m[90m should throw an error, if removing a
non-existing owner[0m
    [32m ✓[0m[90m should remove an owner[0m
[0m    replaceOwner[0m
    [32m ✓[0m[90m should throw an error, if called
directly[0m
    [32m ✓[0m[90m should throw an error, if called by not
an owner[0m
    [32m ✓[0m[90m should throw an error, if replacing a
non-existing owner[0m
    [32m ✓[0m[90m should replace an owner[0m
[0m    changeRequirement[0m
    [32m ✓[0m[90m should throw an error, if called
directly[0m
    [32m ✓[0m[90m should throw an error, if called by not
an owner[0m
    [32m ✓[0m[90m should decrease requirement by 1[0m
[0m    with 4 owners and requirement of 1[0m
[0m    addOwner[0m
    [32m ✓[0m[90m should throw an error, if called
directly[0m
    [32m ✓[0m[90m should throw an error, if called by not
an owner[0m

```



```

[32m ✓[0m[90m should throw an error, if adding an
empty owner[0m
[32m ✓[0m[90m should throw an error, if adding an
existing owner[0m
[32m ✓[0m[90m should add an owner[0m
[0m      removeOwner[0m
[32m ✓[0m[90m should throw an error, if called
directly[0m
[32m ✓[0m[90m should throw an error, if called by not
an owner[0m
[32m ✓[0m[90m should throw an error, if removing a
non-existing owner[0m
[32m ✓[0m[90m should remove an owner[0m
[0m      replaceOwner[0m
[32m ✓[0m[90m should throw an error, if called
directly[0m
[32m ✓[0m[90m should throw an error, if called by not
an owner[0m
[32m ✓[0m[90m should throw an error, if replacing a
non-existing owner[0m
[32m ✓[0m[90m should replace an owner[0m
[0m      changeRequirement[0m
[32m ✓[0m[90m should throw an error, if called
directly[0m
[32m ✓[0m[90m should throw an error, if called by not
an owner[0m
[32m ✓[0m[90m should increase requirement by 1[0m
[0m      with 4 owners and requirement of 2[0m
[0m      addOwner[0m
[32m ✓[0m[90m should throw an error, if called
directly[0m
[32m ✓[0m[90m should throw an error, if called by not
an owner[0m
[32m ✓[0m[90m should throw an error, if adding an
empty owner[0m
[32m ✓[0m[90m should throw an error, if adding an
existing owner[0m
[32m ✓[0m[90m should add an owner[0m
[0m      removeOwner[0m
[32m ✓[0m[90m should throw an error, if called
directly[0m
[32m ✓[0m[90m should throw an error, if called by not
an owner[0m

```

```

[32m ✓[0m[90m should throw an error, if removing a
non-existing owner[0m
[32m ✓[0m[90m should remove an owner[0m
[0m         replaceOwner[0m
[32m ✓[0m[90m should throw an error, if called
directly[0m
[32m ✓[0m[90m should throw an error, if called by not
an owner[0m
[32m ✓[0m[90m should throw an error, if replacing a
non-existing owner[0m
[32m ✓[0m[90m should replace an owner[0m
[0m         changeRequirement[0m
[32m ✓[0m[90m should throw an error, if called
directly[0m
[32m ✓[0m[90m should throw an error, if called by not
an owner[0m
[32m ✓[0m[90m should increase requirement by 1[0m
[0m         with 4 owners and requirement of 3[0m
[0m         addOwner[0m
[32m ✓[0m[90m should throw an error, if called
directly[0m
[32m ✓[0m[90m should throw an error, if called by not
an owner[0m
[32m ✓[0m[90m should throw an error, if adding an
empty owner[0m
[32m ✓[0m[90m should throw an error, if adding an
existing owner[0m
[32m ✓[0m[90m should add an owner[0m
[0m         removeOwner[0m
[32m ✓[0m[90m should throw an error, if called
directly[0m
[32m ✓[0m[90m should throw an error, if called by not
an owner[0m
[32m ✓[0m[90m should throw an error, if removing a
non-existing owner[0m
[32m ✓[0m[90m should remove an owner[0m
[0m         replaceOwner[0m
[32m ✓[0m[90m should throw an error, if called
directly[0m
[32m ✓[0m[90m should throw an error, if called by not
an owner[0m
[32m ✓[0m[90m should throw an error, if replacing a
non-existing owner[0m

```

```

[32m ✓[0m[90m should replace an owner[0m
[0m      changeRequirement[0m
[32m ✓[0m[90m should throw an error, if called
directly[0m
[32m ✓[0m[90m should throw an error, if called by not
an owner[0m
[32m ✓[0m[90m should increase requirement by 1[0m
[0m      with 4 owners and requirement of 4[0m
[0m      addOwner[0m
[32m ✓[0m[90m should throw an error, if called
directly[0m
[32m ✓[0m[90m should throw an error, if called by not
an owner[0m
[32m ✓[0m[90m should throw an error, if adding an
empty owner[0m
[32m ✓[0m[90m should throw an error, if adding an
existing owner[0m
[32m ✓[0m[90m should add an owner[0m
[0m      removeOwner[0m
[32m ✓[0m[90m should throw an error, if called
directly[0m
[32m ✓[0m[90m should throw an error, if called by not
an owner[0m
[32m ✓[0m[90m should throw an error, if removing a
non-existing owner[0m
[32m ✓[0m[90m should remove an owner[0m
[0m      replaceOwner[0m
[32m ✓[0m[90m should throw an error, if called
directly[0m
[32m ✓[0m[90m should throw an error, if called by not
an owner[0m
[32m ✓[0m[90m should throw an error, if replacing a
non-existing owner[0m
[32m ✓[0m[90m should replace an owner[0m
[0m      changeRequirement[0m
[32m ✓[0m[90m should throw an error, if called
directly[0m
[32m ✓[0m[90m should throw an error, if called by not
an owner[0m
[32m ✓[0m[90m should decrease requirement by 1[0m
[0m      with 5 owners and requirement of 3[0m
[0m      addOwner[0m
[32m ✓[0m[90m should throw an error, if called

```

```

directly[0m
    [32m ✓[0m[90m should throw an error, if called by not
an owner[0m
    [32m ✓[0m[90m should throw an error, if adding an
empty owner[0m
    [32m ✓[0m[90m should throw an error, if adding an
existing owner[0m
    [32m ✓[0m[90m should add an owner[0m
[0m        removeOwner[0m
    [32m ✓[0m[90m should throw an error, if called
directly[0m
    [32m ✓[0m[90m should throw an error, if called by not
an owner[0m
    [32m ✓[0m[90m should throw an error, if removing a
non-existing owner[0m
    [32m ✓[0m[90m should remove an owner[0m
[0m        replaceOwner[0m
    [32m ✓[0m[90m should throw an error, if called
directly[0m
    [32m ✓[0m[90m should throw an error, if called by not
an owner[0m
    [32m ✓[0m[90m should throw an error, if replacing a
non-existing owner[0m
    [32m ✓[0m[90m should replace an owner[0m
[0m        changeRequirement[0m
    [32m ✓[0m[90m should throw an error, if called
directly[0m
    [32m ✓[0m[90m should throw an error, if called by not
an owner[0m
    [32m ✓[0m[90m should increase requirement by 1[0m
[0m        events[0m
    [32m ✓[0m[90m should emit deposit event when calling
fallback function[0m
    [32m ✓[0m[90m should emit events when submitting
transaction[0m
    [32m ✓[0m[90m should emit events when confirming
transaction[0m
    [32m ✓[0m[90m should emit events when revoking
confirmation[0m
    [32m ✓[0m[90m should emit events when replacing owner[0m
    [32m ✓[0m[90m should emit events when changing
requirements[0m
    [32m ✓[0m[90m should emit events when execution fails[0m

```

```

    [32m  ✓[0m[90m should emit correct transaction IDs when
submitting multiple transactions[0m

[0m  Contract: OrbsToken[0m
[0m    construction[0m
[0m      invalid arguments[0m
    [32m  ✓[0m[90m should not allow to initialize with a 0
distributor[0m
[0m    success[0m
    [32m  ✓[0m[90m should return correct name after
construction[0m
    [32m  ✓[0m[90m should return correct symbol after
construction[0m
    [32m  ✓[0m[90m should return correct decimal points
after construction[0m
    [32m  ✓[0m[90m should return correct initial totalSupply
after construction[0m
    [32m  ✓[0m[90m should transfer the total supply to the
distributor[0m

[92m [0m[32m 312 passing[0m[90m (3m)[0m

cleaning up

```

## Subscription project

```

Starting our own ganache instance
Compiling ./contracts/DateTime.sol...
Compiling ./contracts/Migrations.sol...
Compiling ./contracts/OrbsToken.sol...
Compiling ./contracts/SubscriptionBilling.sol...
Compiling ./test/DateTimeWrapper.sol...
Compiling ./test/OrbsTokenMock.sol...
Compiling ./test/SubscriptionBillingMock.sol...
Compiling zeppelin-solidity/contracts/math/SafeMath.sol...
Compiling zeppelin-
solidity/contracts/ownership/CanReclaimToken.sol...
Compiling zeppelin-
solidity/contracts/ownership/HasNoContracts.sol...
Compiling zeppelin-
solidity/contracts/ownership/HasNoTokens.sol...
Compiling zeppelin-solidity/contracts/ownership/Ownable.sol...
Compiling zeppelin-
solidity/contracts/token/ERC20/BasicToken.sol...

```

```

Compiling zeppelin-solidity/contracts/token/ERC20/ERC20.sol...
Compiling zeppelin-
solidity/contracts/token/ERC20/ERC20Basic.sol...
Compiling zeppelin-
solidity/contracts/token/ERC20/SafeERC20.sol...
Compiling zeppelin-
solidity/contracts/token/ERC20/StandardToken.sol...

```

Compilation warnings encountered:

```

zeppelin-solidity/contracts/ownership/Ownable.sol:20:3:
Warning: Defining constructors as functions with the same name
as the contract is deprecated. Use "constructor(...) { ... }"
instead.

```

```

    function Ownable() public {
      ^ (Relevant source part starts here and spans across
multiple lines).
,zeppelin-solidity/contracts/ownership/HasNoTokens.sol:21:3:
Warning: Function state mutability can be restricted to pure
    function tokenFallback(address from_, uint256 value_, bytes
data_) external {
      ^ (Relevant source part starts here and spans across
multiple lines).

```

```

[0m[0m
[0m  Contract: DateTime[0m
[0m    getDay[0m
[32m  ✓[0m[90m should return 1 for 0[0m
[32m  ✓[0m[90m should return 31 for 2678399[0m
[32m  ✓[0m[90m should return 1 for 2678400[0m
[32m  ✓[0m[90m should return 28 for 5097599[0m
[32m  ✓[0m[90m should return 1 for 5097600[0m
[32m  ✓[0m[90m should return 31 for 7775999[0m
[32m  ✓[0m[90m should return 1 for 7776000[0m
[32m  ✓[0m[90m should return 30 for 10367999[0m
[32m  ✓[0m[90m should return 1 for 10368000[0m
[32m  ✓[0m[90m should return 31 for 13046399[0m
[32m  ✓[0m[90m should return 1 for 13046400[0m
[32m  ✓[0m[90m should return 30 for 15638399[0m
[32m  ✓[0m[90m should return 1 for 15638400[0m
[32m  ✓[0m[90m should return 31 for 18316799[0m
[32m  ✓[0m[90m should return 1 for 18316800[0m
[32m  ✓[0m[90m should return 31 for 20995199[0m

```

```

[32m  ✓[0m[90m should return 1 for 20995200[0m
[32m  ✓[0m[90m should return 30 for 23587199[0m
[32m  ✓[0m[90m should return 1 for 23587200[0m
[32m  ✓[0m[90m should return 31 for 26265599[0m
[32m  ✓[0m[90m should return 1 for 26265600[0m
[32m  ✓[0m[90m should return 30 for 28857599[0m
[32m  ✓[0m[90m should return 1 for 28857600[0m
[32m  ✓[0m[90m should return 31 for 31535999[0m
[32m  ✓[0m[90m should return 1 for 31536000[0m
[32m  ✓[0m[90m should return 31 for 63071999[0m
[32m  ✓[0m[90m should return 1 for 63072000[0m
[32m  ✓[0m[90m should return 31 for 65750399[0m
[32m  ✓[0m[90m should return 1 for 65750400[0m
[32m  ✓[0m[90m should return 29 for 68255999[0m
[32m  ✓[0m[90m should return 1 for 68256000[0m
[32m  ✓[0m[90m should return 31 for 70934399[0m
[32m  ✓[0m[90m should return 1 for 70934400[0m
[32m  ✓[0m[90m should return 30 for 73526399[0m
[32m  ✓[0m[90m should return 1 for 73526400[0m
[32m  ✓[0m[90m should return 31 for 76204799[0m
[32m  ✓[0m[90m should return 1 for 76204800[0m
[32m  ✓[0m[90m should return 30 for 78796799[0m
[32m  ✓[0m[90m should return 1 for 78796800[0m
[32m  ✓[0m[90m should return 31 for 81475199[0m
[32m  ✓[0m[90m should return 1 for 81475200[0m
[32m  ✓[0m[90m should return 31 for 84153599[0m
[32m  ✓[0m[90m should return 1 for 84153600[0m
[32m  ✓[0m[90m should return 30 for 86745599[0m
[32m  ✓[0m[90m should return 1 for 86745600[0m
[32m  ✓[0m[90m should return 31 for 89423999[0m
[32m  ✓[0m[90m should return 1 for 89424000[0m
[32m  ✓[0m[90m should return 30 for 92015999[0m
[32m  ✓[0m[90m should return 1 for 92016000[0m
[32m  ✓[0m[90m should return 31 for 94694399[0m
[32m  ✓[0m[90m should return 1 for 94694400[0m
[0m    getHour[0m
[32m  ✓[0m[90m should return 23 for 63071999[0m
[32m  ✓[0m[90m should return 0 for 63072000[0m
[32m  ✓[0m[90m should return 0 for 63075599[0m

```

[32m ✓[0m[90m should return 1 for 63075600[0m  
[32m ✓[0m[90m should return 1 for 63079199[0m  
[32m ✓[0m[90m should return 2 for 63079200[0m  
[32m ✓[0m[90m should return 2 for 63082799[0m  
[32m ✓[0m[90m should return 3 for 63082800[0m  
[32m ✓[0m[90m should return 3 for 63086399[0m  
[32m ✓[0m[90m should return 4 for 63086400[0m  
[32m ✓[0m[90m should return 4 for 63089999[0m  
[32m ✓[0m[90m should return 5 for 63090000[0m  
[32m ✓[0m[90m should return 5 for 63093599[0m  
[32m ✓[0m[90m should return 6 for 63093600[0m  
[32m ✓[0m[90m should return 6 for 63097199[0m  
[32m ✓[0m[90m should return 7 for 63097200[0m  
[32m ✓[0m[90m should return 7 for 63100799[0m  
[32m ✓[0m[90m should return 8 for 63100800[0m  
[32m ✓[0m[90m should return 8 for 63104399[0m  
[32m ✓[0m[90m should return 9 for 63104400[0m  
[32m ✓[0m[90m should return 9 for 63107999[0m  
[32m ✓[0m[90m should return 10 for 63108000[0m  
[32m ✓[0m[90m should return 10 for 63111599[0m  
[32m ✓[0m[90m should return 11 for 63111600[0m  
[32m ✓[0m[90m should return 11 for 63115199[0m  
[32m ✓[0m[90m should return 12 for 63115200[0m  
[32m ✓[0m[90m should return 12 for 63118799[0m  
[32m ✓[0m[90m should return 13 for 63118800[0m  
[32m ✓[0m[90m should return 13 for 63122399[0m  
[32m ✓[0m[90m should return 14 for 63122400[0m  
[32m ✓[0m[90m should return 14 for 63125999[0m  
[32m ✓[0m[90m should return 15 for 63126000[0m  
[32m ✓[0m[90m should return 15 for 63129599[0m  
[32m ✓[0m[90m should return 16 for 63129600[0m  
[32m ✓[0m[90m should return 16 for 63133199[0m  
[32m ✓[0m[90m should return 17 for 63133200[0m  
[32m ✓[0m[90m should return 17 for 63136799[0m  
[32m ✓[0m[90m should return 18 for 63136800[0m  
[32m ✓[0m[90m should return 18 for 63140399[0m  
[32m ✓[0m[90m should return 19 for 63140400[0m  
[32m ✓[0m[90m should return 19 for 63143999[0m  
[32m ✓[0m[90m should return 20 for 63144000[0m



```

[32m  ✓[0m[90m should return 20 for 63147599[0m
[32m  ✓[0m[90m should return 21 for 63147600[0m
[32m  ✓[0m[90m should return 21 for 63151199[0m
[32m  ✓[0m[90m should return 22 for 63151200[0m
[32m  ✓[0m[90m should return 22 for 63154799[0m
[32m  ✓[0m[90m should return 23 for 63154800[0m
[32m  ✓[0m[90m should return 0 for 63158400[0m
[0m    isLeapYear[0m
[32m  ✓[0m[90m 1970 should not be a leap year[0m
[32m  ✓[0m[90m 1971 should not be a leap year[0m
[32m  ✓[0m[90m 1972 should be a leap year[0m
[32m  ✓[0m[90m 1973 should not be a leap year[0m
[32m  ✓[0m[90m 1974 should not be a leap year[0m
[32m  ✓[0m[90m 1975 should not be a leap year[0m
[32m  ✓[0m[90m 1976 should be a leap year[0m
[32m  ✓[0m[90m 1977 should not be a leap year[0m
[32m  ✓[0m[90m 1978 should not be a leap year[0m
[32m  ✓[0m[90m 1979 should not be a leap year[0m
[32m  ✓[0m[90m 1980 should be a leap year[0m
[32m  ✓[0m[90m 1981 should not be a leap year[0m
[32m  ✓[0m[90m 1982 should not be a leap year[0m
[32m  ✓[0m[90m 1983 should not be a leap year[0m
[32m  ✓[0m[90m 1984 should be a leap year[0m
[32m  ✓[0m[90m 1985 should not be a leap year[0m
[32m  ✓[0m[90m 1986 should not be a leap year[0m
[32m  ✓[0m[90m 1987 should not be a leap year[0m
[32m  ✓[0m[90m 1988 should be a leap year[0m
[32m  ✓[0m[90m 1989 should not be a leap year[0m
[32m  ✓[0m[90m 1990 should not be a leap year[0m
[32m  ✓[0m[90m 1991 should not be a leap year[0m
[32m  ✓[0m[90m 1992 should be a leap year[0m
[32m  ✓[0m[90m 1993 should not be a leap year[0m
[32m  ✓[0m[90m 1994 should not be a leap year[0m
[32m  ✓[0m[90m 1995 should not be a leap year[0m
[32m  ✓[0m[90m 1996 should be a leap year[0m
[32m  ✓[0m[90m 1997 should not be a leap year[0m
[32m  ✓[0m[90m 1998 should not be a leap year[0m
[32m  ✓[0m[90m 1999 should not be a leap year[0m
[32m  ✓[0m[90m 2000 should be a leap year[0m

```

```
[32m  ✓[0m[90m 2001 should not be a leap year[0m
[32m  ✓[0m[90m 2002 should not be a leap year[0m
[32m  ✓[0m[90m 2003 should not be a leap year[0m
[32m  ✓[0m[90m 2004 should be a leap year[0m
[32m  ✓[0m[90m 2005 should not be a leap year[0m
[32m  ✓[0m[90m 2006 should not be a leap year[0m
[32m  ✓[0m[90m 2007 should not be a leap year[0m
[32m  ✓[0m[90m 2008 should be a leap year[0m
[32m  ✓[0m[90m 2009 should not be a leap year[0m
[32m  ✓[0m[90m 2010 should not be a leap year[0m
[32m  ✓[0m[90m 2011 should not be a leap year[0m
[32m  ✓[0m[90m 2012 should be a leap year[0m
[32m  ✓[0m[90m 2013 should not be a leap year[0m
[32m  ✓[0m[90m 2014 should not be a leap year[0m
[32m  ✓[0m[90m 2015 should not be a leap year[0m
[32m  ✓[0m[90m 2016 should be a leap year[0m
[32m  ✓[0m[90m 2017 should not be a leap year[0m
[32m  ✓[0m[90m 2018 should not be a leap year[0m
[32m  ✓[0m[90m 2019 should not be a leap year[0m
[32m  ✓[0m[90m 2020 should be a leap year[0m
[32m  ✓[0m[90m 2021 should not be a leap year[0m
[32m  ✓[0m[90m 2022 should not be a leap year[0m
[32m  ✓[0m[90m 2023 should not be a leap year[0m
[32m  ✓[0m[90m 2024 should be a leap year[0m
[32m  ✓[0m[90m 2025 should not be a leap year[0m
[32m  ✓[0m[90m 2026 should not be a leap year[0m
[32m  ✓[0m[90m 2027 should not be a leap year[0m
[32m  ✓[0m[90m 2028 should be a leap year[0m
[32m  ✓[0m[90m 2029 should not be a leap year[0m
[32m  ✓[0m[90m 2030 should not be a leap year[0m
[32m  ✓[0m[90m 2031 should not be a leap year[0m
[32m  ✓[0m[90m 2032 should be a leap year[0m
[32m  ✓[0m[90m 2033 should not be a leap year[0m
[32m  ✓[0m[90m 2034 should not be a leap year[0m
[32m  ✓[0m[90m 2035 should not be a leap year[0m
[32m  ✓[0m[90m 2036 should be a leap year[0m
[32m  ✓[0m[90m 2037 should not be a leap year[0m
[32m  ✓[0m[90m 2038 should not be a leap year[0m
[32m  ✓[0m[90m 2039 should not be a leap year[0m
```

```
[32m  ✓[0m[90m 2040 should be a leap year[0m
[32m  ✓[0m[90m 2041 should not be a leap year[0m
[32m  ✓[0m[90m 2042 should not be a leap year[0m
[32m  ✓[0m[90m 2043 should not be a leap year[0m
[32m  ✓[0m[90m 2044 should be a leap year[0m
[32m  ✓[0m[90m 2045 should not be a leap year[0m
[32m  ✓[0m[90m 2046 should not be a leap year[0m
[32m  ✓[0m[90m 2047 should not be a leap year[0m
[32m  ✓[0m[90m 2048 should be a leap year[0m
[32m  ✓[0m[90m 2049 should not be a leap year[0m
[32m  ✓[0m[90m 2050 should not be a leap year[0m
[32m  ✓[0m[90m 2051 should not be a leap year[0m
[32m  ✓[0m[90m 2052 should be a leap year[0m
[32m  ✓[0m[90m 2053 should not be a leap year[0m
[32m  ✓[0m[90m 2054 should not be a leap year[0m
[32m  ✓[0m[90m 2055 should not be a leap year[0m
[32m  ✓[0m[90m 2056 should be a leap year[0m
[32m  ✓[0m[90m 2057 should not be a leap year[0m
[32m  ✓[0m[90m 2058 should not be a leap year[0m
[32m  ✓[0m[90m 2059 should not be a leap year[0m
[32m  ✓[0m[90m 2060 should be a leap year[0m
[32m  ✓[0m[90m 2061 should not be a leap year[0m
[32m  ✓[0m[90m 2062 should not be a leap year[0m
[32m  ✓[0m[90m 2063 should not be a leap year[0m
[32m  ✓[0m[90m 2064 should be a leap year[0m
[32m  ✓[0m[90m 2065 should not be a leap year[0m
[32m  ✓[0m[90m 2066 should not be a leap year[0m
[32m  ✓[0m[90m 2067 should not be a leap year[0m
[32m  ✓[0m[90m 2068 should be a leap year[0m
[32m  ✓[0m[90m 2069 should not be a leap year[0m
[32m  ✓[0m[90m 2070 should not be a leap year[0m
[32m  ✓[0m[90m 2071 should not be a leap year[0m
[32m  ✓[0m[90m 2072 should be a leap year[0m
[32m  ✓[0m[90m 2073 should not be a leap year[0m
[32m  ✓[0m[90m 2074 should not be a leap year[0m
[32m  ✓[0m[90m 2075 should not be a leap year[0m
[32m  ✓[0m[90m 2076 should be a leap year[0m
[32m  ✓[0m[90m 2077 should not be a leap year[0m
[32m  ✓[0m[90m 2078 should not be a leap year[0m
```

```
[32m  ✓[0m[90m 2079 should not be a leap year[0m
[32m  ✓[0m[90m 2080 should be a leap year[0m
[32m  ✓[0m[90m 2081 should not be a leap year[0m
[32m  ✓[0m[90m 2082 should not be a leap year[0m
[32m  ✓[0m[90m 2083 should not be a leap year[0m
[32m  ✓[0m[90m 2084 should be a leap year[0m
[32m  ✓[0m[90m 2085 should not be a leap year[0m
[32m  ✓[0m[90m 2086 should not be a leap year[0m
[32m  ✓[0m[90m 2087 should not be a leap year[0m
[32m  ✓[0m[90m 2088 should be a leap year[0m
[32m  ✓[0m[90m 2089 should not be a leap year[0m
[32m  ✓[0m[90m 2090 should not be a leap year[0m
[32m  ✓[0m[90m 2091 should not be a leap year[0m
[32m  ✓[0m[90m 2092 should be a leap year[0m
[32m  ✓[0m[90m 2093 should not be a leap year[0m
[32m  ✓[0m[90m 2094 should not be a leap year[0m
[32m  ✓[0m[90m 2095 should not be a leap year[0m
[32m  ✓[0m[90m 2096 should be a leap year[0m
[32m  ✓[0m[90m 2097 should not be a leap year[0m
[32m  ✓[0m[90m 2098 should not be a leap year[0m
[32m  ✓[0m[90m 2099 should not be a leap year[0m
[32m  ✓[0m[90m 2100 should not be a leap year[0m
[32m  ✓[0m[90m 2101 should not be a leap year[0m
[32m  ✓[0m[90m 2102 should not be a leap year[0m
[32m  ✓[0m[90m 2103 should not be a leap year[0m
[32m  ✓[0m[90m 2104 should be a leap year[0m
[32m  ✓[0m[90m 2105 should not be a leap year[0m
[32m  ✓[0m[90m 2106 should not be a leap year[0m
[32m  ✓[0m[90m 2107 should not be a leap year[0m
[32m  ✓[0m[90m 2108 should be a leap year[0m
[32m  ✓[0m[90m 2109 should not be a leap year[0m
[32m  ✓[0m[90m 2110 should not be a leap year[0m
[32m  ✓[0m[90m 2111 should not be a leap year[0m
[32m  ✓[0m[90m 2112 should be a leap year[0m
[32m  ✓[0m[90m 2113 should not be a leap year[0m
[32m  ✓[0m[90m 2114 should not be a leap year[0m
[32m  ✓[0m[90m 2115 should not be a leap year[0m
[32m  ✓[0m[90m 2116 should be a leap year[0m
[32m  ✓[0m[90m 2117 should not be a leap year[0m
```

[32m ✓[0m[90m 2118 should not be a leap year[0m  
[32m ✓[0m[90m 2119 should not be a leap year[0m  
[32m ✓[0m[90m 2120 should be a leap year[0m  
[32m ✓[0m[90m 2121 should not be a leap year[0m  
[32m ✓[0m[90m 2122 should not be a leap year[0m  
[32m ✓[0m[90m 2123 should not be a leap year[0m  
[32m ✓[0m[90m 2124 should be a leap year[0m  
[32m ✓[0m[90m 2125 should not be a leap year[0m  
[32m ✓[0m[90m 2126 should not be a leap year[0m  
[32m ✓[0m[90m 2127 should not be a leap year[0m  
[32m ✓[0m[90m 2128 should be a leap year[0m  
[32m ✓[0m[90m 2129 should not be a leap year[0m  
[32m ✓[0m[90m 2130 should not be a leap year[0m  
[32m ✓[0m[90m 2131 should not be a leap year[0m  
[32m ✓[0m[90m 2132 should be a leap year[0m  
[32m ✓[0m[90m 2133 should not be a leap year[0m  
[32m ✓[0m[90m 2134 should not be a leap year[0m  
[32m ✓[0m[90m 2135 should not be a leap year[0m  
[32m ✓[0m[90m 2136 should be a leap year[0m  
[32m ✓[0m[90m 2137 should not be a leap year[0m  
[32m ✓[0m[90m 2138 should not be a leap year[0m  
[32m ✓[0m[90m 2139 should not be a leap year[0m  
[32m ✓[0m[90m 2140 should be a leap year[0m  
[32m ✓[0m[90m 2141 should not be a leap year[0m  
[32m ✓[0m[90m 2142 should not be a leap year[0m  
[32m ✓[0m[90m 2143 should not be a leap year[0m  
[32m ✓[0m[90m 2144 should be a leap year[0m  
[32m ✓[0m[90m 2145 should not be a leap year[0m  
[32m ✓[0m[90m 2146 should not be a leap year[0m  
[32m ✓[0m[90m 2147 should not be a leap year[0m  
[32m ✓[0m[90m 2148 should be a leap year[0m  
[32m ✓[0m[90m 2149 should not be a leap year[0m  
[32m ✓[0m[90m 2150 should not be a leap year[0m  
[32m ✓[0m[90m 2151 should not be a leap year[0m  
[32m ✓[0m[90m 2152 should be a leap year[0m  
[32m ✓[0m[90m 2153 should not be a leap year[0m  
[32m ✓[0m[90m 2154 should not be a leap year[0m  
[32m ✓[0m[90m 2155 should not be a leap year[0m  
[32m ✓[0m[90m 2156 should be a leap year[0m

[32m ✓[0m[90m 2157 should not be a leap year[0m  
[32m ✓[0m[90m 2158 should not be a leap year[0m  
[32m ✓[0m[90m 2159 should not be a leap year[0m  
[32m ✓[0m[90m 2160 should be a leap year[0m  
[32m ✓[0m[90m 2161 should not be a leap year[0m  
[32m ✓[0m[90m 2162 should not be a leap year[0m  
[32m ✓[0m[90m 2163 should not be a leap year[0m  
[32m ✓[0m[90m 2164 should be a leap year[0m  
[32m ✓[0m[90m 2165 should not be a leap year[0m  
[32m ✓[0m[90m 2166 should not be a leap year[0m  
[32m ✓[0m[90m 2167 should not be a leap year[0m  
[32m ✓[0m[90m 2168 should be a leap year[0m  
[32m ✓[0m[90m 2169 should not be a leap year[0m  
[32m ✓[0m[90m 2170 should not be a leap year[0m  
[32m ✓[0m[90m 2171 should not be a leap year[0m  
[32m ✓[0m[90m 2172 should be a leap year[0m  
[32m ✓[0m[90m 2173 should not be a leap year[0m  
[32m ✓[0m[90m 2174 should not be a leap year[0m  
[32m ✓[0m[90m 2175 should not be a leap year[0m  
[32m ✓[0m[90m 2176 should be a leap year[0m  
[32m ✓[0m[90m 2177 should not be a leap year[0m  
[32m ✓[0m[90m 2178 should not be a leap year[0m  
[32m ✓[0m[90m 2179 should not be a leap year[0m  
[32m ✓[0m[90m 2180 should be a leap year[0m  
[32m ✓[0m[90m 2181 should not be a leap year[0m  
[32m ✓[0m[90m 2182 should not be a leap year[0m  
[32m ✓[0m[90m 2183 should not be a leap year[0m  
[32m ✓[0m[90m 2184 should be a leap year[0m  
[32m ✓[0m[90m 2185 should not be a leap year[0m  
[32m ✓[0m[90m 2186 should not be a leap year[0m  
[32m ✓[0m[90m 2187 should not be a leap year[0m  
[32m ✓[0m[90m 2188 should be a leap year[0m  
[32m ✓[0m[90m 2189 should not be a leap year[0m  
[32m ✓[0m[90m 2190 should not be a leap year[0m  
[32m ✓[0m[90m 2191 should not be a leap year[0m  
[32m ✓[0m[90m 2192 should be a leap year[0m  
[32m ✓[0m[90m 2193 should not be a leap year[0m  
[32m ✓[0m[90m 2194 should not be a leap year[0m  
[32m ✓[0m[90m 2195 should not be a leap year[0m

```

[32m  ✓[0m[90m 2196 should be a leap year[0m
[32m  ✓[0m[90m 2197 should not be a leap year[0m
[32m  ✓[0m[90m 2198 should not be a leap year[0m
[32m  ✓[0m[90m 2199 should not be a leap year[0m
[32m  ✓[0m[90m 2200 should not be a leap year[0m
[32m  ✓[0m[90m 2201 should not be a leap year[0m
[32m  ✓[0m[90m 2202 should not be a leap year[0m
[32m  ✓[0m[90m 2203 should not be a leap year[0m
[0m    getMinute[0m
[32m  ✓[0m[90m should return 59 for 63071999[0m
[32m  ✓[0m[90m should return 0 for 63072000[0m
[32m  ✓[0m[90m should return 0 for 63072059[0m
[32m  ✓[0m[90m should return 1 for 63072060[0m
[32m  ✓[0m[90m should return 1 for 63072119[0m
[32m  ✓[0m[90m should return 2 for 63072120[0m
[32m  ✓[0m[90m should return 2 for 63072179[0m
[32m  ✓[0m[90m should return 3 for 63072180[0m
[32m  ✓[0m[90m should return 3 for 63072239[0m
[32m  ✓[0m[90m should return 4 for 63072240[0m
[32m  ✓[0m[90m should return 4 for 63072299[0m
[32m  ✓[0m[90m should return 5 for 63072300[0m
[32m  ✓[0m[90m should return 5 for 63072359[0m
[32m  ✓[0m[90m should return 6 for 63072360[0m
[32m  ✓[0m[90m should return 6 for 63072419[0m
[32m  ✓[0m[90m should return 7 for 63072420[0m
[32m  ✓[0m[90m should return 7 for 63072479[0m
[32m  ✓[0m[90m should return 8 for 63072480[0m
[32m  ✓[0m[90m should return 8 for 63072539[0m
[32m  ✓[0m[90m should return 9 for 63072540[0m
[32m  ✓[0m[90m should return 9 for 63072599[0m
[32m  ✓[0m[90m should return 10 for 63072600[0m
[32m  ✓[0m[90m should return 10 for 63072659[0m
[32m  ✓[0m[90m should return 11 for 63072660[0m
[32m  ✓[0m[90m should return 11 for 63072719[0m
[32m  ✓[0m[90m should return 12 for 63072720[0m
[32m  ✓[0m[90m should return 12 for 63072779[0m
[32m  ✓[0m[90m should return 13 for 63072780[0m
[32m  ✓[0m[90m should return 13 for 63072839[0m
[32m  ✓[0m[90m should return 14 for 63072840[0m

```

[32m ✓[0m[90m should return 14 for 63072899[0m  
[32m ✓[0m[90m should return 15 for 63072900[0m  
[32m ✓[0m[90m should return 15 for 63072959[0m  
[32m ✓[0m[90m should return 16 for 63072960[0m  
[32m ✓[0m[90m should return 16 for 63073019[0m  
[32m ✓[0m[90m should return 17 for 63073020[0m  
[32m ✓[0m[90m should return 17 for 63073079[0m  
[32m ✓[0m[90m should return 18 for 63073080[0m  
[32m ✓[0m[90m should return 18 for 63073139[0m  
[32m ✓[0m[90m should return 19 for 63073140[0m  
[32m ✓[0m[90m should return 19 for 63073199[0m  
[32m ✓[0m[90m should return 20 for 63073200[0m  
[32m ✓[0m[90m should return 20 for 63073259[0m  
[32m ✓[0m[90m should return 21 for 63073260[0m  
[32m ✓[0m[90m should return 21 for 63073319[0m  
[32m ✓[0m[90m should return 22 for 63073320[0m  
[32m ✓[0m[90m should return 22 for 63073379[0m  
[32m ✓[0m[90m should return 23 for 63073380[0m  
[32m ✓[0m[90m should return 23 for 63073439[0m  
[32m ✓[0m[90m should return 24 for 63073440[0m  
[32m ✓[0m[90m should return 24 for 63073499[0m  
[32m ✓[0m[90m should return 25 for 63073500[0m  
[32m ✓[0m[90m should return 25 for 63073559[0m  
[32m ✓[0m[90m should return 26 for 63073560[0m  
[32m ✓[0m[90m should return 26 for 63073619[0m  
[32m ✓[0m[90m should return 27 for 63073620[0m  
[32m ✓[0m[90m should return 27 for 63073679[0m  
[32m ✓[0m[90m should return 28 for 63073680[0m  
[32m ✓[0m[90m should return 28 for 63073739[0m  
[32m ✓[0m[90m should return 29 for 63073740[0m  
[32m ✓[0m[90m should return 29 for 63073799[0m  
[32m ✓[0m[90m should return 30 for 63073800[0m  
[32m ✓[0m[90m should return 30 for 63073859[0m  
[32m ✓[0m[90m should return 31 for 63073860[0m  
[32m ✓[0m[90m should return 31 for 63073919[0m  
[32m ✓[0m[90m should return 32 for 63073920[0m  
[32m ✓[0m[90m should return 32 for 63073979[0m  
[32m ✓[0m[90m should return 33 for 63073980[0m  
[32m ✓[0m[90m should return 33 for 63074039[0m



[32m ✓[0m[90m should return 34 for 63074040[0m  
[32m ✓[0m[90m should return 34 for 63074099[0m  
[32m ✓[0m[90m should return 35 for 63074100[0m  
[32m ✓[0m[90m should return 35 for 63074159[0m  
[32m ✓[0m[90m should return 36 for 63074160[0m  
[32m ✓[0m[90m should return 36 for 63074219[0m  
[32m ✓[0m[90m should return 37 for 63074220[0m  
[32m ✓[0m[90m should return 37 for 63074279[0m  
[32m ✓[0m[90m should return 38 for 63074280[0m  
[32m ✓[0m[90m should return 38 for 63074339[0m  
[32m ✓[0m[90m should return 39 for 63074340[0m  
[32m ✓[0m[90m should return 39 for 63074399[0m  
[32m ✓[0m[90m should return 40 for 63074400[0m  
[32m ✓[0m[90m should return 40 for 63074459[0m  
[32m ✓[0m[90m should return 41 for 63074460[0m  
[32m ✓[0m[90m should return 41 for 63074519[0m  
[32m ✓[0m[90m should return 42 for 63074520[0m  
[32m ✓[0m[90m should return 42 for 63074579[0m  
[32m ✓[0m[90m should return 43 for 63074580[0m  
[32m ✓[0m[90m should return 43 for 63074639[0m  
[32m ✓[0m[90m should return 44 for 63074640[0m  
[32m ✓[0m[90m should return 44 for 63074699[0m  
[32m ✓[0m[90m should return 45 for 63074700[0m  
[32m ✓[0m[90m should return 45 for 63074759[0m  
[32m ✓[0m[90m should return 46 for 63074760[0m  
[32m ✓[0m[90m should return 46 for 63074819[0m  
[32m ✓[0m[90m should return 47 for 63074820[0m  
[32m ✓[0m[90m should return 47 for 63074879[0m  
[32m ✓[0m[90m should return 48 for 63074880[0m  
[32m ✓[0m[90m should return 48 for 63074939[0m  
[32m ✓[0m[90m should return 49 for 63074940[0m  
[32m ✓[0m[90m should return 49 for 63074999[0m  
[32m ✓[0m[90m should return 50 for 63075000[0m  
[32m ✓[0m[90m should return 50 for 63075059[0m  
[32m ✓[0m[90m should return 51 for 63075060[0m  
[32m ✓[0m[90m should return 51 for 63075119[0m  
[32m ✓[0m[90m should return 52 for 63075120[0m  
[32m ✓[0m[90m should return 52 for 63075179[0m  
[32m ✓[0m[90m should return 53 for 63075180[0m

```

[32m  ✓[0m[90m should return 53 for 63075239[0m
[32m  ✓[0m[90m should return 54 for 63075240[0m
[32m  ✓[0m[90m should return 54 for 63075299[0m
[32m  ✓[0m[90m should return 55 for 63075300[0m
[32m  ✓[0m[90m should return 55 for 63075359[0m
[32m  ✓[0m[90m should return 56 for 63075360[0m
[32m  ✓[0m[90m should return 56 for 63075419[0m
[32m  ✓[0m[90m should return 57 for 63075420[0m
[32m  ✓[0m[90m should return 57 for 63075479[0m
[32m  ✓[0m[90m should return 58 for 63075480[0m
[32m  ✓[0m[90m should return 58 for 63075539[0m
[32m  ✓[0m[90m should return 59 for 63075540[0m
[32m  ✓[0m[90m should return 59 for 63075599[0m
[0m    getMonth[0m
[32m  ✓[0m[90m should return 1 for 0[0m
[32m  ✓[0m[90m should return 1 for 2678399[0m
[32m  ✓[0m[90m should return 2 for 2678400[0m
[32m  ✓[0m[90m should return 2 for 5097599[0m
[32m  ✓[0m[90m should return 3 for 5097600[0m
[32m  ✓[0m[90m should return 3 for 7775999[0m
[32m  ✓[0m[90m should return 4 for 7776000[0m
[32m  ✓[0m[90m should return 4 for 10367999[0m
[32m  ✓[0m[90m should return 5 for 10368000[0m
[32m  ✓[0m[90m should return 5 for 13046399[0m
[32m  ✓[0m[90m should return 6 for 13046400[0m
[32m  ✓[0m[90m should return 6 for 15638399[0m
[32m  ✓[0m[90m should return 7 for 15638400[0m
[32m  ✓[0m[90m should return 7 for 18316799[0m
[32m  ✓[0m[90m should return 8 for 18316800[0m
[32m  ✓[0m[90m should return 8 for 20995199[0m
[32m  ✓[0m[90m should return 9 for 20995200[0m
[32m  ✓[0m[90m should return 9 for 23587199[0m
[32m  ✓[0m[90m should return 10 for 23587200[0m
[32m  ✓[0m[90m should return 10 for 26265599[0m
[32m  ✓[0m[90m should return 11 for 26265600[0m
[32m  ✓[0m[90m should return 11 for 28857599[0m
[32m  ✓[0m[90m should return 12 for 28857600[0m
[32m  ✓[0m[90m should return 12 for 31535999[0m
[32m  ✓[0m[90m should return 1 for 31536000[0m

```

```

[32m  ✓[0m[90m should return 12 for 63071999[0m
[32m  ✓[0m[90m should return 1 for 63072000[0m
[32m  ✓[0m[90m should return 1 for 65750399[0m
[32m  ✓[0m[90m should return 2 for 65750400[0m
[32m  ✓[0m[90m should return 2 for 68255999[0m
[32m  ✓[0m[90m should return 3 for 68256000[0m
[32m  ✓[0m[90m should return 3 for 70934399[0m
[32m  ✓[0m[90m should return 4 for 70934400[0m
[32m  ✓[0m[90m should return 4 for 73526399[0m
[32m  ✓[0m[90m should return 5 for 73526400[0m
[32m  ✓[0m[90m should return 5 for 76204799[0m
[32m  ✓[0m[90m should return 6 for 76204800[0m
[32m  ✓[0m[90m should return 6 for 78796799[0m
[32m  ✓[0m[90m should return 7 for 78796800[0m
[32m  ✓[0m[90m should return 7 for 81475199[0m
[32m  ✓[0m[90m should return 8 for 81475200[0m
[32m  ✓[0m[90m should return 8 for 84153599[0m
[32m  ✓[0m[90m should return 9 for 84153600[0m
[32m  ✓[0m[90m should return 9 for 86745599[0m
[32m  ✓[0m[90m should return 10 for 86745600[0m
[32m  ✓[0m[90m should return 10 for 89423999[0m
[32m  ✓[0m[90m should
return 11 for 89424000[0m
[32m  ✓[0m[90m should return 11 for 92015999[0m
[32m  ✓[0m[90m should return 12 for 92016000[0m
[32m  ✓[0m[90m should return 12 for 94694399[0m
[32m  ✓[0m[90m should return 1 for 94694400[0m
[0m    getSecond[0m
[32m  ✓[0m[90m should return 59 for 63071999[0m
[32m  ✓[0m[90m should return 0 for 63072000[0m
[32m  ✓[0m[90m should return 1 for 63072001[0m
[32m  ✓[0m[90m should return 2 for 63072002[0m
[32m  ✓[0m[90m should return 3 for 63072003[0m
[32m  ✓[0m[90m should return 4 for 63072004[0m
[32m  ✓[0m[90m should return 5 for 63072005[0m
[32m  ✓[0m[90m should return 6 for 63072006[0m
[32m  ✓[0m[90m should return 7 for 63072007[0m
[32m  ✓[0m[90m should return 8 for 63072008[0m
[32m  ✓[0m[90m should return 9 for 63072009[0m

```

[32m ✓[0m[90m should return 10 for 63072010[0m  
[32m ✓[0m[90m should return 11 for 63072011[0m  
[32m ✓[0m[90m should return 12 for 63072012[0m  
[32m ✓[0m[90m should return 13 for 63072013[0m  
[32m ✓[0m[90m should return 14 for 63072014[0m  
[32m ✓[0m[90m should return 15 for 63072015[0m  
[32m ✓[0m[90m should return 16 for 63072016[0m  
[32m ✓[0m[90m should return 17 for 63072017[0m  
[32m ✓[0m[90m should return 18 for 63072018[0m  
[32m ✓[0m[90m should return 19 for 63072019[0m  
[32m ✓[0m[90m should return 20 for 63072020[0m  
[32m ✓[0m[90m should return 21 for 63072021[0m  
[32m ✓[0m[90m should return 22 for 63072022[0m  
[32m ✓[0m[90m should return 23 for 63072023[0m  
[32m ✓[0m[90m should return 24 for 63072024[0m  
[32m ✓[0m[90m should return 25 for 63072025[0m  
[32m ✓[0m[90m should return 26 for 63072026[0m  
[32m ✓[0m[90m should return 27 for 63072027[0m  
[32m ✓[0m[90m should return 28 for 63072028[0m  
[32m ✓[0m[90m should return 29 for 63072029[0m  
[32m ✓[0m[90m should return 30 for 63072030[0m  
[32m ✓[0m[90m should return 31 for 63072031[0m  
[32m ✓[0m[90m should return 32 for 63072032[0m  
[32m ✓[0m[90m should return 33 for 63072033[0m  
[32m ✓[0m[90m should return 34 for 63072034[0m  
[32m ✓[0m[90m should return 35 for 63072035[0m  
[32m ✓[0m[90m should return 36 for 63072036[0m  
[32m ✓[0m[90m should return 37 for 63072037[0m  
[32m ✓[0m[90m should return 38 for 63072038[0m  
[32m ✓[0m[90m should return 39 for 63072039[0m  
[32m ✓[0m[90m should return 40 for 63072040[0m  
[32m ✓[0m[90m should return 41 for 63072041[0m  
[32m ✓[0m[90m should return 42 for 63072042[0m  
[32m ✓[0m[90m should return 43 for 63072043[0m  
[32m ✓[0m[90m should return 44 for 63072044[0m  
[32m ✓[0m[90m should return 45 for 63072045[0m  
[32m ✓[0m[90m should return 46 for 63072046[0m  
[32m ✓[0m[90m should return 47 for 63072047[0m  
[32m ✓[0m[90m should return 48 for 63072048[0m

```

[32m  ✓[0m[90m should return 49 for 63072049[0m
[32m  ✓[0m[90m should return 50 for 63072050[0m
[32m  ✓[0m[90m should return 51 for 63072051[0m
[32m  ✓[0m[90m should return 52 for 63072052[0m
[32m  ✓[0m[90m should return 53 for 63072053[0m
[32m  ✓[0m[90m should return 54 for 63072054[0m
[32m  ✓[0m[90m should return 55 for 63072055[0m
[32m  ✓[0m[90m should return 56 for 63072056[0m
[32m  ✓[0m[90m should return 57 for 63072057[0m
[32m  ✓[0m[90m should return 58 for 63072058[0m
[32m  ✓[0m[90m should return 59 for 63072059[0m
[32m  ✓[0m[90m should return 0 for 63072060[0m
[0m    getBeginningOfMonth[0m
[32m  ✓[0m[90m should return 0 for
{"year":1970,"month":1}[0m
[32m  ✓[0m[90m should return 28857600 for
{"year":1970,"month":12}[0m
[32m  ✓[0m[90m should return 1522540800 for
{"year":2018,"month":4}[0m
[32m  ✓[0m[90m should return 917827200 for
{"year":1999,"month":2}[0m
[32m  ✓[0m[90m should return 1541030400 for
{"year":2018,"month":11}[0m
[0m    getNextMonth[0m
[32m  ✓[0m[90m should return {"year":1970,"month":2} for
{"year":1970,"month":1}[0m
[32m  ✓[0m[90m should return {"year":1971,"month":1} for
{"year":1970,"month":12}[0m
[32m  ✓[0m[90m should return {"year":2018,"month":5} for
{"year":2018,"month":4}[0m
[32m  ✓[0m[90m should return {"year":1989,"month":3} for
{"year":1989,"month":2}[0m
[32m  ✓[0m[90m should return {"year":2019,"month":1} for
{"year":2018,"month":12}[0m
[0m    toTimeStamp[0m
[32m  ✓[0m[90m should return 0 for
{"year":1970,"month":1,"day":1,"hours":0,"minutes":0,"seconds":0}[0m
[32m  ✓[0m[90m should return 31535999 for
{"year":1970,"month":12,"day":31,"hours":23,"minutes":59,"seconds":59}[0m

```

```

    [32m  ✓[0m[90m should return 31536000 for
{"year":1971,"month":1,"day":1,"hours":0,"minutes":0,"seconds"
:0}[0m
    [32m  ✓[0m[90m should return 31536000 for
{"year":1971,"month":1,"day":0,"hours":0,"minutes":0,"seconds"
:0}[0m
    [32m  ✓[0m[90m should return 1456790399 for
{"year":2016,"month":2,"day":29,"hours":23,"minutes":59,"secon
ds":59}[0m
    [32m  ✓[0m[90m should return 1456790400 for
{"year":2016,"month":3,"day":1,"hours":0,"minutes":0,"seconds"
:0}[0m
[0m    getWeekday[0m
    [32m  ✓[0m[90m should return 3 for 67737599[0m
    [32m  ✓[0m[90m should return 4 for 67737600[0m
    [32m  ✓[0m[90m should return 4 for 67823999[0m
    [32m  ✓[0m[90m should return 5 for 67824000[0m
    [32m  ✓[0m[90m should return 5 for 67910399[0m
    [32m  ✓[0m[90m should return 6 for 67910400[0m
    [32m  ✓[0m[90m should return 6 for 67996799[0m
    [32m  ✓[0m[90m should return 0 for 67996800[0m
    [32m  ✓[0m[90m should return 0 for 68083199[0m
    [32m  ✓[0m[90m should return 1 for 68083200[0m
    [32m  ✓[0m[90m should return 1 for 68169599[0m
    [32m  ✓[0m[90m should return 2 for 68169600[0m
    [32m  ✓[0m[90m should return 2 for 68255999[0m
    [32m  ✓[0m[90m should return 3 for 68256000[0m
    [32m  ✓[0m[90m should return 3 for 68342399[0m
    [32m  ✓[0m[90m should return 4 for 68342400[0m
[0m    getWeekday[0m
    [32m  ✓[0m[90m should return 1970 for 0[0m
    [32m  ✓[0m[90m should return 1971 for 31536000[0m
    [32m  ✓[0m[90m should return 1970 for 31535999[0m
    [32m  ✓[0m[90m should return 1972 for 63072000[0m
    [32m  ✓[0m[90m should return 1971 for 63071999[0m
    [32m  ✓[0m[90m should return 1973 for 94694400[0m
    [32m  ✓[0m[90m should return 1972 for 94694399[0m
    [32m  ✓[0m[90m should return 1974 for 126230400[0m
    [32m  ✓[0m[90m should return 1973 for 126230399[0m
    [32m  ✓[0m[90m should return 1975 for 157766400[0m

```

```

[32m  ✓[0m[90m should return 1974 for 157766399[0m
[32m  ✓[0m[90m should return 1976 for 189302400[0m
[32m  ✓[0m[90m should return 1975 for 189302399[0m
[32m  ✓[0m[90m should return 1977 for 220924800[0m
[32m  ✓[0m[90m should return 1976 for 220924799[0m
[32m  ✓[0m[90m should return 1978 for 252460800[0m
[32m  ✓[0m[90m should return 1977 for 252460799[0m
[32m  ✓[0m[90m should return 1979 for 283996800[0m
[32m  ✓[0m[90m should return 1978 for 283996799[0m
[32m  ✓[0m[90m should return 1980 for 315532800[0m
[32m  ✓[0m[90m should return 1979 for 315532799[0m
[32m  ✓[0m[90m should return 1981 for 347155200[0m
[32m  ✓[0m[90m should return 1980 for 347155199[0m
[32m  ✓[0m[90m should return 1982 for 378691200[0m
[32m  ✓[0m[90m should return 1981 for 378691199[0m
[32m  ✓[0m[90m should return 1983 for 410227200[0m
[32m  ✓[0m[90m should return 1982 for 410227199[0m
[32m  ✓[0m[90m should return 1984 for 441763200[0m
[32m  ✓[0m[90m should return 1983 for 441763199[0m
[32m  ✓[0m[90m should return 1985 for 473385600[0m
[32m  ✓[0m[90m should return 1984 for 473385599[0m
[32m  ✓[0m[90m should return 1986 for 504921600[0m
[32m  ✓[0m[90m should return 1985 for 504921599[0m
[32m  ✓[0m[90m should return 1987 for 536457600[0m
[32m  ✓[0m[90m should return 1986 for 536457599[0m
[32m  ✓[0m[90m should return 1988 for 567993600[0m
[32m  ✓[0m[90m should return 1987 for 567993599[0m
[32m  ✓[0m[90m should return 1989 for 599616000[0m
[32m  ✓[0m[90m should return 1988 for 599615999[0m
[32m  ✓[0m[90m should return 1990 for 631152000[0m
[32m  ✓[0m[90m should return 1989 for 631151999[0m
[32m  ✓[0m[90m should return 1991 for 662688000[0m
[32m  ✓[0m[90m should return 1990 for 662687999[0m
[32m  ✓[0m[90m should return 1992 for 694224000[0m
[32m  ✓[0m[90m should return 1991 for 694223999[0m
[32m  ✓[0m[90m should return 1993 for 725846400[0m
[32m  ✓[0m[90m should return 1992 for 725846399[0m
[32m  ✓[0m[90m should return 1994 for 757382400[0m
[32m  ✓[0m[90m should return 1993 for 757382399[0m

```

```

[32m  ✓[0m[90m should return 1995 for 788918400[0m
[32m  ✓[0m[90m should return 1994 for 788918399[0m
[32m  ✓[0m[90m should return 1996 for 820454400[0m
[32m  ✓[0m[90m should return 1995 for 820454399[0m
[32m  ✓[0m[90m should return 1997 for 852076800[0m
[32m  ✓[0m[90m should return 1996 for 852076799[0m
[32m  ✓[0m[90m should return 1998 for 883612800[0m
[32m  ✓[0m[90m should return 1997 for 883612799[0m
[32m  ✓[0m[90m should return 1999 for 915148800[0m
[32m  ✓[0m[90m should return 1998 for 915148799[0m
[32m  ✓[0m[90m should return 2000 for 946684800[0m
[32m  ✓[0m[90m should return 1999 for 946684799[0m
[32m  ✓[0m[90m should return 2001 for 978307200[0m
[32m  ✓[0m[90m should return 2000 for 978307199[0m
[32m  ✓[0m[90m should return 2002 for 1009843200[0m
[32m  ✓[0m[90m should return 2001 for 1009843199[0m
[32m  ✓[0m[90m should return 2003 for 1041379200[0m
[32m  ✓[0m[90m should return 2002 for 1041379199[0m
[32m  ✓[0m[90m should return 2004 for 1072915200[0m
[32m  ✓[0m[90m should return 2003 for 1072915199[0m
[32m  ✓[0m[90m should return 2005 for 1104537600[0m
[32m  ✓[0m[90m should return 2004 for 1104537599[0m
[32m  ✓[0m[90m should return 2006 for 1136073600[0m
[32m  ✓[0m[90m should return 2005 for 1136073599[0m
[32m  ✓[0m[90m should return 2007 for 1167609600[0m
[32m  ✓[0m[90m should return 2006 for 1167609599[0m
[32m  ✓[0m[90m should return 2008 for 1199145600[0m
[32m  ✓[0m[90m should return 2007 for 1199145599[0m
[32m  ✓[0m[90m should return 2009 for 1230768000[0m
[32m  ✓[0m[90m should return 2008 for 1230767999[0m
[32m  ✓[0m[90m should return 2010 for 1262304000[0m
[32m  ✓[0m[90m should return 2009 for 1262303999[0m
[32m  ✓[0m[90m should return 2011 for 1293840000[0m
[32m  ✓[0m[90m should return 2010 for 1293839999[0m
[32m  ✓[0m[90m should return 2012 for 1325376000[0m
[32m  ✓[0m[90m should return 2011 for 1325375999[0m
[32m  ✓[0m[90m should return 2013 for 1356998400[0m
[32m  ✓[0m[90m should return 2012 for 1356998399[0m
[32m  ✓[0m[90m should return 2014 for 1388534400[0m

```



```

[32m  ✓[0m[90m should return 2013 for 1388534399[0m
[32m  ✓[0m[90m should return 2015 for 1420070400[0m
[32m  ✓[0m[90m should return 2014 for 1420070399[0m
[32m  ✓[0m[90m should return 2016 for 1451606400[0m
[32m  ✓[0m[90m should return 2015 for 1451606399[0m
[32m  ✓[0m[90m should return 2017 for 1483228800[0m
[32m  ✓[0m[90m should return 2016 for 1483228799[0m
[32m  ✓[0m[90m should return 2018 for 1514764800[0m
[32m  ✓[0m[90m should return 2017 for 1514764799[0m
[32m  ✓[0m[90m should return 2019 for 1546300800[0m
[32m  ✓[0m[90m should return 2018 for 1546300799[0m
[32m  ✓[0m[90m should return 2020 for 1577836800[0m

[0m  Contract: SubscriptionBilling[0m
[0m    construction[0m
      [32m  ✓[0m[90m should not allow to initialize with a null
token[0m
      [32m  ✓[0m[90m should not allow to initialize with an
empty array of federation members[0m
      [32m  ✓[0m[90m should not allow to initialize with too
many federation members[0m
      [32m  ✓[0m[90m should not allow to initialize with
duplicate federation members[0m
      [32m  ✓[0m[90m should not allow to initialize with a 0
minimal subscription allocation[0m
      [32m  ✓[0m[90m should correctly initialize the minimal
monthly subscription[0m
      [32m  ✓[0m[90m should report version[0m
[0m    subscription and fees[0m
[0m      with 1 federation members[0m
      [32m  ✓[0m[90m should error when called with an empty
id[0m
      [32m  ✓[0m[90m should error when called with an empty
profile[0m
      [32m  ✓[0m[90m should error when called with no
tokens[0m
      [32m  ✓[0m[90m should error when called with not enough
tokens[0m
      [32m  ✓[0m[90m should error when called without prior
funding[0m
      [32m  ✓[0m[90m should error when called with more tokens
than the user has[0m

```

```

[32m ✓[0m[90m should error when directly called with a
past date[0m

[32m ✓[0m[90m should error when fetching current
monthly subscription with an empty id[0m

[32m ✓[0m[90m should error when fetching monthly
subscription with an empty id[0m

[32m ✓[0m[90m should not allow to distribute future
fees[0m
[0m         this month[0m

[32m ✓[0m[90m should subscribe to a single
subscription[0m

[32m ✓[0m[90m should subscribe to multiple
subscriptions[0m

[32m ✓[0m[90m should be able to top-up the
subscription[0m

[32m ✓[0m[90m should not be able to change the
profile during top-up of the subscription[0m

[32m ✓[0m[90m should distribute the fees between the
federation members[0m
[0m         next month[0m

[32m ✓[0m[90m should subscribe to a single
subscription[0m

[32m ✓[0m[90m should subscribe to multiple
subscriptions[0m

[32m ✓[0m[90m should not be able to change the
profile during top-up of the subscription[0m

[32m ✓[0m[90m should be able to top-up the
subscription[0m

[32m ✓[0m[90m should distribute the fees between the
federation members[0m
[0m         during 3 months[0m

[32m ✓[0m[90m should distribute the fees between the
federation members[0m
[0m         with 2 federation members[0m

[32m ✓[0m[90m should error when called with an empty
id[0m

[32m ✓[0m[90m should error when called with an empty
profile[0m

[32m ✓[0m[90m should error when called with no
tokens[0m

[32m ✓[0m[90m should error when called with not enough
tokens[0m

```

```

[32m  ✓[0m[90m should error when called without prior
funding[0m
[32m  ✓[0m[90m should error when called with more tokens
than the user has[0m
[32m  ✓[0m[90m should error when directly called with a
past date[0m
[32m  ✓[0m[90m should error when fetching current
monthly subscription with an empty id[0m
[32m  ✓[0m[90m should error when fetching monthly
subscription with an empty id[0m
[32m  ✓[0m[90m should not allow to distribute future
fees[0m
[0m      this month[0m
[32m  ✓[0m[90m should subscribe to a single
subscription[0m
[32m  ✓[0m[90m should subscribe to multiple
subscriptions[0m
[32m  ✓[0m[90m should be able to top-up the
subscription[0m
[32m  ✓[0m[90m should not be able to change the
profile during top-up of the subscription[0m
[32m  ✓[0m[90m should distribute the fees between the
federation members[0m
[0m      next month[0m
[32m  ✓[0m[90m should subscribe to a single
subscription[0m
[32m  ✓[0m[90m should subscribe to multiple
subscriptions[0m
[32m  ✓[0m[90m should not be able to change the
profile during top-up of the subscription[0m
[32m  ✓[0m[90m should be able to top-up the
subscription[0m
[32m  ✓[0m[90m should distribute the fees between the
federation members[0m
[0m      during 3 months[0m
[32m  ✓[0m[90m should distribute the fees between the
federation members[0m
[0m      with 3 federation members[0m
[32m  ✓[0m[90m should error when called with an empty
id[0m
[32m  ✓[0m[90m should error when called with an empty
profile[0m

```

```

[32m  ✓[0m[90m should error when called with no
tokens[0m
[32m  ✓[0m[90m should error when called with not enough
tokens[0m
[32m  ✓[0m[90m should error when called without prior
funding[0m
[32m  ✓[0m[90m should error when called with more tokens
than the user has[0m
[32m  ✓[0m[90m should error when directly called with a
past date[0m
[32m  ✓[0m[90m should error when fetching current
monthly subscription with an empty id[0m
[32m  ✓[0m[90m should error when fetching monthly
subscription with an empty id[0m
[32m  ✓[0m[90m should not allow to distribute future
fees[0m
[0m      this month[0m
[32m  ✓[0m[90m should subscribe to a single
subscription[0m
[32m  ✓[0m[90m should subscribe to multiple
subscriptions[0m
[32m  ✓[0m[90m should be able to top-up the
subscription[0m
[32m  ✓[0m[90m should not be able to change the
profile during top-up of the subscription[0m
[32m  ✓[0m[90m should distribute the fees between the
federation members[0m
[0m      next month[0m
[32m  ✓[0m[90m should subscribe to a single
subscription[0m
[32m  ✓[0m[90m should subscribe to multiple
subscriptions[0m
[32m  ✓[0m[90m should not be able to change the
profile during top-up of the subscription[0m
[32m  ✓[0m[90m should be able to top-up the
subscription[0m
[32m  ✓[0m[90m should distribute the fees between the
federation members[0m
[0m      during 3 months[0m
[32m  ✓[0m[90m should distribute the fees between the
federation members[0m
[0m      with 4 federation members[0m

```

```

[32m  ✓[0m[90m should error when called with an empty
id[0m
[32m  ✓[0m[90m should error when called with an empty
profile[0m
[32m  ✓[0m[90m should error when called with no
tokens[0m
[32m  ✓[0m[90m should error when called with not enough
tokens[0m
[32m  ✓[0m[90m should error when called without prior
funding[0m
[32m  ✓[0m[90m should error when called with more tokens
than the user has[0m
[32m  ✓[0m[90m should error when directly called with a
past date[0m
[32m  ✓[0m[90m should error when fetching current
monthly subscription with an empty id[0m
[32m  ✓[0m[90m should error when fetching monthly
subscription with an empty id[0m
[32m  ✓[0m[90m should not allow to distribute future
fees[0m
[0m      this month[0m
[32m  ✓[0m[90m should subscribe to a single
subscription[0m
[32m  ✓[0m[90m should subscribe to multiple
subscriptions[0m
[32m  ✓[0m[90m should be able to top-up the
subscription[0m
[32m  ✓[0m[90m should not be able to change the
profile during top-up of the subscription[0m
[32m  ✓[0m[90m should distribute the fees between the
federation members[0m
[0m      next month[0m
[32m  ✓[0m[90m should subscribe to a single
subscription[0m
[32m  ✓[0m[90m should subscribe to multiple
subscriptions[0m
[32m  ✓[0m[90m should not be able to change the
profile during top-up of the subscription[0m
[32m  ✓[0m[90m should be able to top-up the
subscription[0m
[32m  ✓[0m[90m should distribute the fees between the
federation members[0m
[0m      during 3 months[0m

```

```

[32m ✓[0m[90m should distribute the fees between the
federation members[0m
[0m      with 5 federation members[0m
[32m ✓[0m[90m should error when called with an empty
id[0m
[32m ✓[0m[90m should error when called with an empty
profile[0m
[32m ✓[0m[90m should error when called with no
tokens[0m
[32m ✓[0m[90m should error when called with not enough
tokens[0m
[32m ✓[0m[90m should error when called without prior
funding[0m
[32m ✓[0m[90m should error when called with more tokens
than the user has[0m
[32m ✓[0m[90m should error when directly called with a
past date[0m
[32m ✓[0m[90m should error when fetching current
monthly subscription with an empty id[0m
[32m ✓[0m[90m should error when fetching monthly
subscription with an empty id[0m
[32m ✓[0m[90m should not allow to distribute future
fees[0m
[0m      this month[0m
[32m ✓[0m[90m should subscribe to a single
subscription[0m
[32m ✓[0m[90m should subscribe to multiple
subscriptions[0m
[32m ✓[0m[90m should be able to top-up the
subscription[0m
[32m ✓[0m[90m should not be able to change the
profile during top-up of the subscription[0m
[32m ✓[0m[90m should distribute the fees between the
federation members[0m
[0m      next month[0m
[32m ✓[0m[90m should subscribe to a single
subscription[0m
[32m ✓[0m[90m should subscribe to multiple
subscriptions[0m
[32m ✓[0m[90m should not be able to change the
profile during top-up of the subscription[0m
[32m ✓[0m[90m should be able to top-up the
subscription[0m

```

```

[32m ✓[0m[90m should distribute the fees between the
federation members[0m
[0m         during 3 months[0m
[32m ✓[0m[90m should distribute the fees between the
federation members[0m
[0m         with 6 federation members[0m
[32m ✓[0m[90m should error when called with an empty
id[0m
[32m ✓[0m[90m should error when called with an empty
profile[0m
[32m ✓[0m[90m should error when called with no
tokens[0m
[32m ✓[0m[90m should error when called with not enough
tokens[0m
[32m ✓[0m[90m should error when called without prior
funding[0m
[32m ✓[0m[90m should error when called with more tokens
than the user has[0m
[32m ✓[0m[90m should error when directly called with a
past date[0m
[32m ✓[0m[90m should error when fetching current
monthly subscription with an empty id[0m
[32m ✓[0m[90m should error when fetching monthly
subscription with an empty id[0m
[32m ✓[0m[90m should not allow to distribute future
fees[0m
[0m         this month[0m
[32m ✓[0m[90m should subscribe to a single
subscription[0m
[32m ✓[0m[90m should subscribe to multiple
subscriptions[0m
[32m ✓[0m[90m should be able to top-up the
subscription[0m
[32m ✓[0m[90m should not be able to change the
profile during top-up of the subscription[0m
[32m ✓[0m[90m should distribute the fees between the
federation members[0m
[0m         next month[0m
[32m ✓[0m[90m should subscribe to a single
subscription[0m
[32m ✓[0m[90m should subscribe to multiple
subscriptions[0m

```

[32m ✓[0m[90m should not be able to change the profile during top-up of the subscription[0m

[32m ✓[0m[90m should be able to top-up the subscription[0m

[32m ✓[0m[90m should distribute the fees between the federation members[0m  
[0m during 3 months[0m

[32m ✓[0m[90m should distribute the fees between the federation members[0m  
[0m with 20 federation members[0m

[32m ✓[0m[90m should error when called with an empty id[0m

[32m ✓[0m[90m should error when called with an empty profile[0m

[32m ✓[0m[90m should error when called with no tokens[0m

[32m ✓[0m[90m should error when called with not enough tokens[0m

[32m ✓[0m[90m should error when called without prior funding[0m

[32m ✓[0m[90m should error when called with more tokens than the user has[0m

[32m ✓[0m[90m should error when directly called with a past date[0m

[32m ✓[0m[90m should error when fetching current monthly subscription with an empty id[0m

[32m ✓[0m[90m should error when fetching monthly subscription with an empty id[0m

[32m ✓[0m[90m should not allow to distribute future fees[0m  
[0m this month[0m

[32m ✓[0m[90m should subscribe to a single subscription[0m

[32m ✓[0m[90m should subscribe to multiple subscriptions[0m

[32m ✓[0m[90m should be able to top-up the subscription[0m

[32m ✓[0m[90m should not be able to change the profile during top-up of the subscription[0m

[32m ✓[0m[90m should distribute the fees between the federation members[0m  
[0m next month[0m



```

[32m ✓[0m[90m should subscribe to a single
subscription[0m
[32m ✓[0m[90m should subscribe to multiple
subscriptions[0m
[32m ✓[0m[90m should not be able to change the
profile during top-up of the subscription[0m
[32m ✓[0m[90m should be able to top-up the
subscription[0m
[32m ✓[0m[90m should distribute the fees between the
federation members[0m
[0m         during 3 months[0m
[32m ✓[0m[90m should distribute the fees between the
federation members[0m
[0m         with 100 federation members[0m
[32m ✓[0m[90m should error when called with an empty
id[0m
[32m ✓[0m[90m should error when called with an empty
profile[0m
[32m ✓[0m[90m should error when called with no
tokens[0m
[32m ✓[0m[90m should error when called with not enough
tokens[0m
[32m ✓[0m[90m should error when called without prior
funding[0m
[32m ✓[0m[90m should error when called with more tokens
than the user has[0m
[32m ✓[0m[90m should error when directly called with a
past date[0m
[32m ✓[0m[90m should error when fetching current
monthly subscription with an empty id[0m
[32m ✓[0m[90m should error when fetching monthly
subscription with an empty id[0m
[32m ✓[0m[90m should not allow to distribute future
fees[0m
[0m         this month[0m
[32m ✓[0m[90m should subscribe to a single
subscription[0m
[32m ✓[0m[90m should subscribe to multiple
subscriptions[0m
[32m ✓[0m[90m should be able to top-up the
subscription[0m
[32m ✓[0m[90m should not be able to change the
profile during top-up of the subscription[0m

```

```

[32m ✓[0m[90m should distribute the fees between the
federation members[0m
[0m      next month[0m
[32m ✓[0m[90m should subscribe to a single
subscription[0m
[32m ✓[0m[90m should subscribe to multiple
subscriptions[0m
[32m ✓[0m[90m should not be able to change the
profile during top-up of the subscription[0m
[32m ✓[0m[90m should be able to top-up the
subscription[0m
[32m ✓[0m[90m should distribute the fees between the
federation members[0m
[0m      during 3 months[0m
[32m ✓[0m[90m should distribute the fees between the
federation members[0m[33m (17890ms)[0m

[92m [0m[32m 875 passing[0m[90m (5m)[0m

cleaning up

```

## Solhint output

```

SubscriptionBilling.sol
139:18  warning  Variable "DistributedFees" is unused
no-unused-vars
139:33  error      Expected indentation of 12 spaces but found
32      indent
139:18  error      Variable name must be in mixedCase
var-name-mixedcase
146:2   error      Line length must be no more than 120 but
current length is 125  max-line-length
149:42  warning    Avoid to make time-based decisions in your
business logic        not-rely-on-time
155:2   error      Line length must be no more than 120 but
current length is 125  max-line-length
174:2   error      Line length must be no more than 120 but
current length is 125  max-line-length
181:31  warning    Avoid to make time-based decisions in your
business logic        not-rely-on-time
210:24  error      Expected indentation of 8 spaces but found
23      indent

```

```

    210:14  error      Variable name must be in mixedCase
var-name-mixedcase
    210:14  warning    Variable "Subscribed" is unused
no-unused-vars
    217:24  warning    Avoid to make time-based decisions in your
business logic      not-rely-on-time

✗ 12 problems (7 errors, 5 warnings)

OrbsToken.sol
    7:1    error      Definition must be surrounded with two blank
line indent  two-lines-top-level-separator
    22:22  error      Expected indentation of 8 spaces but found
21            indent
    22:14  error      Variable name must be in mixedCase
var-name-mixedcase
    22:14  warning    Variable "Transfer" is unused
no-unused-vars

✗ 4 problems (3 errors, 1 warning)

```

## Solium output

```

.\SubscriptionBilling.sol
    120:16  error      Only use indent of 12
spaces.            indentation
    149:41  warning    Avoid using 'now' (alias to
'block.timestamp'). security/no-block-members
    181:30  warning    Avoid using 'now' (alias to
'block.timestamp'). security/no-block-members
    217:23  warning    Avoid using 'now' (alias to
'block.timestamp'). security/no-block-members

✗ 1 error, 3 warnings found.

.\OrbsToken.sol

No issues found.

```