AKIRA TECH

PROJECT

Rysk Dynamic Hedging

CLIENT

Rysk Finance

DATE

August 2022

REVIEWERS

Andrei Simion

@andreiashu

Table of Contents

Details

- **Client** Rysk Finance
- **Date** August 2022
- **Reviewers** Andrei Simion (@andreiashu)
- **Repository**: Rysk Dynamic Hedging
- **Commit hash** `0dd6fbb7f492fc4d569706a9b4b0f2e262d8c086`
- **Technologies**
  - Solidity
  - Typescript

Issues Summary

SEVERITY	OPEN	CLOSED
Informational	1	0
Minor	10	0
Medium	2	0
Major	2	0

Executive summary

This report represents the results of the engagement with **Rysk Finance** to review **Rysk Dynamic Hedging**.

The review was conducted over the course of **3 weeks** from **8th of August to 26th of August, 2022**. A total of **15 person-days** were spent reviewing the code.

Week 1

During the first week, I had a kickoff call with the Rysk team to get familiar with the high-level scope of the project. Then, I reviewed the code in the `LiquidityPool` and any interactions with other contracts or libraries. For example, `Accounting.sol` is part of the

LiquidityPool, but it was moved into a separate contract for better modularity and to avoid Solidity's max contract size.

The Rysk team has provided clear documentation, which helped me better understand their design decisions and the codebase.

Week 2

The second week I spent reviewing contracts adjacent to the `LiquidityPool` and went through the rest of the code in scope for this review.

Week 3

In the last week of the code review, I started passing through the code with a clearer understanding of the overall interactions between contracts (internal and external to the project). However, some of the more intricate flows needed more attention. For example, going through the Hedging Reactor contracts (`UniswapV3HedgingReactor.sol` and `PerpHedgingReactor.sol`) surfaced an issue that would have gotten the `LiquidityPool` stuck at `executeEpochCalculation` in some cases.

Scope

The initial review focused on the Rysk Dynamic Hedging repository, identified by the commit hash `0dd6fbb7f492fc4d569706a9b4b0f2e262d8c086` .

I focused on manually reviewing the codebase, searching for security issues such as, but not limited to, re-entrancy problems, transaction ordering, block timestamp dependency, exception handling, call stack depth limitation, integer overflow/underflow, self-destructible contracts, unsecured balance, use of origin, costly gas patterns, architectural problems, code readability.

Includes:

- code/packages/contracts/contracts/Accounting.sol
- code/packages/contracts/contracts/AlphaOptionHandler.sol
- code/packages/contracts/contracts/PriceFeed.sol
- code/packages/contracts/contracts/LiquidityPool.sol
- code/packages/contracts/contracts/OptionRegistry.sol
- code/packages/contracts/contracts/OptionHandler.sol
- code/packages/contracts/contracts/Protocol.sol
- code/packages/contracts/contracts/Authority.sol
- code/packages/contracts/contracts/libraries/CustomErrors.sol

- code/packages/contracts/contracts/libraries/BlackScholes.sol
- code/packages/contracts/contracts/libraries/OptionsCompute.sol
- code/packages/contracts/contracts/libraries/EnumerableSet.sol
- code/packages/contracts/contracts/libraries/NormalDist.sol
- code/packages/contracts/contracts/libraries/SafeTransferLib.sol
- code/packages/contracts/contracts/libraries/OpynInteractions.sol
- code/packages/contracts/contracts/libraries/Types.sol
- code/packages/contracts/contracts/libraries/AccessControl.sol
- code/packages/contracts/contracts/utils/ReentrancyGuard.sol
- code/packages/contracts/contracts/utils/Volatility.sol
- code/packages/contracts/contracts/hedging/UniswapV3HedgingReactor.sol
- code/packages/contracts/contracts/hedging/PerpHedgingReactor.sol
- code/packages/contracts/contracts/AlphaPortfolioValuesFeed.sol
- code/packages/contracts/contracts/PortfolioValuesFeed.sol
- code/packages/contracts/contracts/tokens/ERC20.sol
- code/packages/contracts/contracts/tokens/WETH.sol
- code/packages/contracts/contracts/tokens/MintableERC20.sol

Does not include

- code/packages/contracts/contracts/VolatilityFeed.sol

Recommendations

I identified a few possible general improvements that are not security issues during the review, which will bring value to the developers and the community reviewing and using the product.

Increase the number of tests

A good rule of thumb is to have 100% test coverage. This does not guarantee the lack of security problems, but it means that the desired functionality behaves as intended. The negative tests also bring a lot of value because not allowing some actions to happen is also part of the desired behavior.

Issues

[LiquidityPool] in some cases the contract can get stuck in the `executeEpochCalculation` call

Status Open Severity Major

Description

The `LiquidityPool` contract specifies a `bufferPercentage` value that represents the amount of minimum `collateralAsset` to keep in the vault when writing new options for margin requirements:

code/packages/contracts/contracts/LiquidityPool.sol#L81-L82

```
	// buffer of funds to not be used to write new options in case of margin requirements (as percenta
	uint256 public bufferPercentage = 2000;
```

This specification is enforced by the use of the `LiquidityPool.checkBuffer` method, which will revert if this condition is not met:

code/packages/contracts/contracts/LiquidityPool.sol#L975-L985

```
	 * @notice calculates amount of liquidity that can be used before hitting buffer
	 * @return bufferRemaining the amount of liquidity available before reaching buffer in e6
	 */
	function checkBuffer() public view returns (uint256 bufferRemaining) {
		// calculate max amount of liquidity pool funds that can be used before reaching max buffe
		uint256 collateralBalance = getBalance(collateralAsset);
		uint256 collateralBuffer = (collateralAllocated * bufferPercentage) / MAX_BPS;
		// revert if buffer allowance already hit
		if (collateralBuffer > collateralBalance) {
			revert CustomErrors.MaxLiquidityBufferReached();
		}
```

During a `hedgeDelta` call on the `UniswapV3HedgingReactor`, the `collateralAsset` balance in the `LiquidityPool` can fall below the `bufferPercentage` value. Specifically: when `delta` is negative, the `UniswapV3HedgingReactor` contract will transfer `collateralAsset` tokens from the `LiquidityPool` to itself to perform a swap from `collateralAsset` to `wETH` tokens:

code/packages/contracts/contracts/hedging/UniswapV3HedgingReactor.sol#L227-L235

```
	function _swapExactOutputSingle(
		uint256 _amountOut,
		uint256 _amountInMaximum,
		address _sellToken
	) internal returns (int256, uint256) {
		if (ILiquidityPool(parentLiquidityPool).getBalance(collateralAsset) < _amountInMaximum) {
			revert CustomErrors.WithdrawExceedsLiquidity();
		}
		SafeTransferLib.safeTransferFrom(_sellToken, msg.sender, address(this), _amountInMaximum);
```

I confirmed with the Rysk team that this is a valid use case. However, when the contract is in a state whereby a Hedging Reactor contract causes the `collateralAsset` balance in the pool to fall below the required buffer, there is an unintended side effect: the `Accounting.executeEpochCalculation` will revert when calling the `checkBuffer` method at line 255:

code/packages/contracts/contracts/Accounting.sol#L254-L255

```
        // get the liquidity that can be withdrawn from the pool without hitting the collateral re
        uint256 bufferRemaining = liquidityPool.checkBuffer();
```

This means that keepers (governor or managers) will not be able to execute a `LiquidityPool.executeEpochCalculation` successfully, and the pool is now in limbo, requiring manual intervention in order for the epoch calculation to be performed.

Recommendation

Remove the revert from `checkBuffer` and handle the condition separately in each calling method. Within the `executeEpochCalculation` method, there is no need to revert if the buffer is not met, for example.

[PriceFeed] `getNormalizedRate` might return outdated or invalid pricing data

Status Open Severity Major

Description

Accurate pricing data feed for different assets used in the system is at the core of running a stable and functionally correct platform.

`PriceFeed.getNormalizedRate` uses Chainlink's V3 interface to fetch and parse pricing feeds:

code/packages/contracts/contracts/PriceFeed.sol#L58-L60

```
        AggregatorV3Interface feed = AggregatorV3Interface(feedAddress);
        uint8 feedDecimals = feed.decimals();
        (, int256 rate, , , ) = feed.latestRoundData();
```

The issue, however, is that data returned from Chainlink can sometimes be stale or invalid, and the code in `getNormalizedRate` does not validate this case.

NB: `PriceFeed.getRate` shares this same issue, but it is only used in the Typescript tests:

code/packages/contracts/contracts/PriceFeed.sol#L46-L51

```
        function getRate(address underlying, address strike) external view returns (uint256) {
                address feedAddress = priceFeeds[underlying][strike];
                require(feedAddress != address(0), "Price feed does not exist");
                AggregatorV3Interface feed = AggregatorV3Interface(feedAddress);
                (, int256 rate, , , ) = feed.latestRoundData();
                return uint256(rate);
```

Recommendation

A complete code to validate data returned by Chainlink's `latestRoundData` would include checking against a few other attributes:

```
            (uint80 roundId, int256 answer, , uint256 timestamp, uint80 answeredInRound ) = aggregator.late
             require(answer > 0, "ChainLinkPricer: price is lower than 0");
             require(timestamp != 0, "ROUND_NOT_COMPLETE");
             require(block.timestamp <= timestamp + stalePriceDelay, "STALE_PRICE");
             require(answeredInRound >= roundId, "STALE_PRICE");
```

The `stalePriceDelay` parameter should be configured on a per-pair basis since different data feeds have various guarantees for how long ago the last answer was committed to the blockchain.

Chainlink's documentation in their v0.7 version of the code, in which some pair data feeds still run on:

```
/**
 * @notice get data about the latest round. Consumers are encouraged to check
 * that they're receiving fresh data by inspecting the updatedAt and
 * answeredInRound return values.
 * Note that different underlying implementations of AggregatorV3Interface
 * have slightly different semantics for some of the return values. Consumers
 * should determine what implementations they expect to receive
 * data from and validate that they can properly handle return data from all
 * of them.
```

[BlackScholes] Some methods might underflow

Status Open Severity Medium

Description

There are several places in the `BlackScholes` contract whereby the return value of a subtraction is cast to an unsigned value without checking that the resulting value does not underflow: `callOptionPriceGreeks`, `putOptionPriceGreeks`, `callOptionPrice`, `putOptionPrice`.

code/packages/contracts/contracts/libraries/BlackScholes.sol#L39

```
                return uint256(priceCdf - strikeBy);
```

code/packages/contracts/contracts/libraries/BlackScholes.sol#L54

```
                quote = uint256(priceCdf - strikeBy);
```

code/packages/contracts/contracts/libraries/BlackScholes.sol#L70

```
                quote = uint256(strikeBy - priceCdf);
```

code/packages/contracts/contracts/libraries/BlackScholes.sol#L86

```
                return uint256(strikeBy - priceCdf);
```

Recommendation

Ensure that an underflow is not possible or create a test that shows that, mathematically, this case can't happen.

[Authority] push methods should validate against `null` address values

Status Open Severity Medium

Description

Logically it doesn't make sense to set to `null` address the `governor`, `manager` or a `guardian`:

code/packages/contracts/contracts/Authority.sol#L41

```
        function pushGovernor(address _newGovernor, bool _effectiveImmediately) external {
```

code/packages/contracts/contracts/Authority.sol#L48

```
        function pushGuardian(address _newGuardian) external {
```

code/packages/contracts/contracts/Authority.sol#L53

```
        function pushManager(address _newManager, bool _effectiveImmediately) external {
```

Recommendation

Add validation against the `null` address value to all the `push` methods in the `Authority` contract.

NB: I set this issue to `Medium` since setting the `governor` variable to the `null` address would render any contracts extending `Authority` to be governor-less, a highly undesirable outcome.

[LiquidityPool] `_collateralAsset` decimals should be validated against `OptionsCompute.SCALE_DECIMALS` in constructor

Status Open Severity Minor

Description

Within the `LiquidityPool` contract there are different points whereby the state variable `collateralAsset` is used to convert to/from a specific number of decimals:

code/packages/contracts/contracts/LiquidityPool.sol#L950

```
                assets = _getNormalizedBalance(collateralAsset) + OptionsCompute.convertFromDecimals(colla
```

Within the `convertFromDecimals` method, the code ensures that the number of decimals passed is not higher than the `SCALE_DECIMALS` constant:

code/packages/contracts/contracts/libraries/OptionsCompute.sol#L30-L33

```
        function convertFromDecimals(uint256 value, uint256 decimals) internal pure returns (uint256) {
                if (decimals > SCALE_DECIMALS) {
                        revert();
                }
```

Because of this, it makes sense to ensure that whatever ERC20 token address is passed as `_collateralAsset` in the constructor, to be validated against the `SCALE_DECIMALS`. In other words: the contract should revert on deployment if `ERC20(_collateralAsset).decimals() > SCALE_DECIMALS`

[LiquidityPool] `removeHedgingReactorAddress` can save on gas

Status Open Severity Minor

Description

Instead of evaluating the result of `hedgingReactors.length - 1` on every iteration of the loop, `removeHedgingReactorAddress` can save on gas by caching the value of `hedgingReactors.length - 1` in a local memory variable:

code/packages/contracts/contracts/LiquidityPool.sol#L220

```
                for (uint256 i = _index; i < hedgingReactors.length - 1; i++) {
```

[LiquidityPool] validation issues

Status Open Severity Minor

Description

There are several methods in `LiquidityPool` contract that would benefit from having their arguments validated.

There is no validation for nil addresses or duplicated reactor addresses in `setHedgingReactorAddress`:

code/packages/contracts/contracts/LiquidityPool.sol#L195-L198

```
        function setHedgingReactorAddress(address _reactorAddress) external {
                _onlyGovernor();


                hedgingReactors.push(_reactorAddress);
                SafeTransferLib.safeApprove(ERC20(collateralAsset), _reactorAddress, type(uint256).max);
```

code/packages/contracts/contracts/LiquidityPool.sol#L283

```
        function setBufferPercentage(uint256 _bufferPercentage) external {
```

code/packages/contracts/contracts/LiquidityPool.sol#L300

```
        function setMaxTimeDeviationThreshold(uint256 _maxTimeDeviationThreshold) external {
```

code/packages/contracts/contracts/LiquidityPool.sol#L231-L238

```
        function setNewOptionParams(
                uint128 _newMinCallStrike,
                uint128 _newMaxCallStrike,
                uint128 _newMinPutStrike,
                uint128 _newMaxPutStrike,
                uint128 _newMinExpiry,
                uint128 _newMaxExpiry
        ) external {
```

[Authority] `pushGuardian` method should emit an event

Status Open Severity Minor

Description

There are no events emitted when a new `guardian` address is added:

code/packages/contracts/contracts/Authority.sol#L48-L51

```
        function pushGuardian(address _newGuardian) external {
                _onlyGovernor();
                guardian[_newGuardian] = true;
        }
```

Other similar methods in the `Authority` contract emit events - this helps external parties to monitor and make use of such events.

[Authority] Confusing `GuardianPulled` event name in `revokeGuardian` method

Status Open Severity Minor

Description

`Authority.revokeGuardian` allows a governor to revoke an address' guardian *role*:

code/packages/contracts/contracts/Authority.sol#L66-L69

```
        function revokeGuardian(address _guardian) external {
                _onlyGovernor();
                emit GuardianPulled(_guardian);
                guardian[_guardian] = false;
```

This action will emit a `GuardianPulled` event. The name of the event is confusing since the `Pulled` suffix is also used by `GovernorPulled` and `ManagerPulled` events to signal that a new governor or manager has taken ownership of their role:

code/packages/contracts/contracts/Authority.sol#L72-L75

```
        function pullManager() external {
                require(msg.sender == newManager, "!newManager");
                emit ManagerPulled(manager, newManager);
                manager = newManager;
```

Recommendation

`GuardianRevoked` might be a better event name to be emitted in the `revokeGuardian` method.

[Authority] When `_effectiveImmediately` flag is true, a different set of events is emitted for the same state outcome

Status Open Severity Minor

Description

A new governor of the contract can be elected by calling `pushGovernor` method:

code/packages/contracts/contracts/Authority.sol#L41-L45

```
    function pushGovernor(address _newGovernor, bool _effectiveImmediately) external {
        _onlyGovernor();
        if (_effectiveImmediately) governor = _newGovernor;
        newGovernor = _newGovernor;
        emit GovernorPushed(governor, newGovernor, _effectiveImmediately);
```

In order for the effect to be immediate, the `_effectiveImmediately` can be set to True. In this case, the outcome of the method is as if `pushGovernor` and `pullGovernor` were called sequentially (by the initial governor and then the new one, respectively):

code/packages/contracts/contracts/Authority.sol#L43-L44

```
        if (_effectiveImmediately) governor = _newGovernor;
        newGovernor = _newGovernor;
```

There are several issues with this logic:

- although `governor` state variable is updated, the `GovernorPulled` event is not emitted;
- `newGovernor` state variable is updated which would enable the address set in `_newGovernor` to call `pullGovernor` successfully - but this is a no-op call that will emit a `GovernorPulled` event - even though the `governor` already had the new value set
- in `pullGovernor` method, the `newGovernor` state variable is not reset to `null` value - thus allowing this method to be called indefinitely and emitting duplicated, no-op, `GovernorPulled` events:

code/packages/contracts/contracts/Authority.sol#L60-L63

```
    function pullGovernor() external {
        require(msg.sender == newGovernor, "!newGovernor");
        emit GovernorPulled(governor, newGovernor);
        governor = newGovernor;
```

Recommendation

When `_effectiveImmediately` flag is true, the code should:

- emit a `GovernorPulled` event
- reset `newGovernor` variable to `null` address

NB: the observations and recommendations are also applicable to `pushManager` and `pullManager` methods.

`Authority.constructor` does not validate addresses against `null` value

Status Open Severity Minor

Description

Passing `null` address values in the constructor of `Authority` contract does not logically make sense:

code/packages/contracts/contracts/Authority.sol#L26-L35

```
    constructor(
        address _governor,
        address _guardian,
        address _manager
    ) AccessControl(IAuthority(address(this))) {
        governor = _governor;
        emit GovernorPushed(address(0), governor, true);
        guardian[_guardian] = true;
        emit GuardianPushed(_guardian, true);
        manager = _manager;
```

Recommendation

Therefore the code should ensure that the `_governor`, `_guardian` and `_manager` variables are not `null`.

[LiquidityPool] Actions that affect `LiquidityPool`'s functionality should emit relevant events

Status Open Severity Minor

Description

Within the `LiquidityPool` contract, there are administrative methods accessible by privileged roles within the platform. One such example is the `pause` method which a guardian or governor can invoke to pause operations on the contract:

code/packages/contracts/contracts/LiquidityPool.sol#L175-L178

```
        function pause() external {
                _onlyGuardian();
                _pause();
        }
```

The `_pause` method is inherited from OpenZeppelin's `Pausable` contract. It will mark the `_paused` state variable as true and **emit a** `Paused` **event**:

```
    function _pause() internal virtual whenNotPaused {
        _paused = true;
        emit Paused(_msgSender());
    }
```

The event emitted helps external parties and stakeholders in the platform monitor activity of different roles in the system.

However, there are different other methods that affect the functionality of the contract which do not emit any events:

code/packages/contracts/contracts/LiquidityPool.sol#L263

```
        function setMaxDiscount(uint256 _maxDiscount) external {
```

code/packages/contracts/contracts/LiquidityPool.sol#L231

```
        function setNewOptionParams(
```

code/packages/contracts/contracts/LiquidityPool.sol#L208

```
        function removeHedgingReactorAddress(uint256 _index, bool _override) external {
```

code/packages/contracts/contracts/LiquidityPool.sol#L195

```
        function setHedgingReactorAddress(address _reactorAddress) external {
```

code/packages/contracts/contracts/LiquidityPool.sol#L180

```
        function pauseUnpauseTrading(bool _pause) external {
```

Recommendation

Add code that emits events for all the methods in the contract that affect its functionality.

`Authority.pushGovernor` emits incorrect values for `GovernorPushed` event

Status Open Severity Minor

Description

`pushGovernor` method allows to elect a new governor:

code/packages/contracts/contracts/Authority.sol#L41-L45

```
        function pushGovernor(address _newGovernor, bool _effectiveImmediately) external {
                _onlyGovernor();
                if (_effectiveImmediately) governor = _newGovernor;
                newGovernor = _newGovernor;
                emit GovernorPushed(governor, newGovernor, _effectiveImmediately);
```

The `_effectiveImmediately` flag can be set to true to allow this change to be in effect immediately.

The issue, however, is that the condition for `_effectiveImmediately` is evaluated *before* the `GovernorPushed` event is emitted. Therefore the `governor` variable will have a new value for the governor:

code/packages/contracts/contracts/Authority.sol#L43-L45

```
                if (_effectiveImmediately) governor = _newGovernor;
                newGovernor = _newGovernor;
                emit GovernorPushed(governor, newGovernor, _effectiveImmediately);
```

Recommendation

Emit the `GovernorPushed` event *before* the `_effectiveImmediately` condition is evaluated.

`Authority.constructor` can save gas by using local instead of state variables for events

Status Open Severity Minor

Description

The `Authority.constructor` emits several events upon deployment:

code/packages/contracts/contracts/Authority.sol#L26-L36

```
        constructor(
                address _governor,
                address _guardian,
                address _manager
        ) AccessControl(IAuthority(address(this))) {
                governor = _governor;
                emit GovernorPushed(address(0), governor, true);
                guardian[_guardian] = true;
                emit GuardianPushed(_guardian, true);
                manager = _manager;
                emit ManagerPushed(address(0), manager, true);
```

Two of these events read from state variables when a local copy is already available:

code/packages/contracts/contracts/Authority.sol#L32

```
                emit GovernorPushed(address(0), governor, true);
```

code/packages/contracts/contracts/Authority.sol#L36

```
                emit ManagerPushed(address(0), manager, true);
```

Recommendation

Pass the `_manager` and `_governor` local variables for emitting events. This will save gas.

[LiquidityPool] `completeWithdraw` should withdraw all shares that the user already redeemed

Status Open Severity Informational

Description

The `completeWithdraw` method allows for a specific number of shares to be withdrawn after a user has already `redeemed` them:

code/packages/contracts/contracts/LiquidityPool.sol#L659

```
        function completeWithdraw(uint256 _shares) external whenNotPaused nonReentrant returns (uint256) {
```

The issue, however, is that the case whereby a user withdraws a less than redeemed number of shares does not make sense in the Rysk platform. This happens because a further call to `initiateWithdraw` would fail in the `Accounting.initiateWithdraw` call. As long as `withdrawalReceipts[msg.sender].shares` is of a non-zero value and `withdrawalReceipts[msg.sender].epoch` is not the current one (`withdrawalEpoch`), a call to `initiateWithdraw` will fail at L179 in `Accounting.sol` contract:

code/packages/contracts/contracts/Accounting.sol#L174-L180

```
                if (withdrawalReceipt.epoch == currentEpoch) {
                        withdrawalShares = existingShares + shares;
                } else {
                        // do 100 wei just in case of any rounding issues
                        if (existingShares > 100) {
                                revert CustomErrors.ExistingWithdrawal();
                        }
```

Recommendation

Remove the `shares` argument from `completeWithdraw` method and always perform a complete withdrawal for all the shares the user has redeemed already.

Artifacts

Surya

Sūrya is a utility tool for smart contract systems. It provides a number of visual outputs and information about the structure of smart contracts. It also supports querying the function call graph in multiple ways to aid in the manual inspection and control flow analysis of contracts.

Sūrya's Description Report

Files Description Table

File Name	
code/packages/contracts/contracts/Accounting.sol	94a5da6
code/packages/contracts/contracts/AlphaOptionHandler.sol	ab07d53
code/packages/contracts/contracts/PriceFeed.sol	da57045
code/packages/contracts/contracts/LiquidityPool.sol	c0e3ba5
code/packages/contracts/contracts/OptionRegistry.sol	4eb1cfff
code/packages/contracts/contracts/OptionHandler.sol	62775ee
code/packages/contracts/contracts/Protocol.sol	41616ec
code/packages/contracts/contracts/Authority.sol	db1b974
code/packages/contracts/contracts/libraries/CustomErrors.sol	1eecdae
code/packages/contracts/contracts/libraries/BlackScholes.sol	9a01e62
code/packages/contracts/contracts/libraries/OptionsCompute.sol	ff93cef1
code/packages/contracts/contracts/libraries/EnumerableSet.sol	21d718c
code/packages/contracts/contracts/libraries/NormalDist.sol	1c9a1d5
code/packages/contracts/contracts/libraries/SafeTransferLib.sol	d37fd6b
code/packages/contracts/contracts/libraries/OpynInteractions.sol	05dccd0
code/packages/contracts/contracts/libraries/Types.sol	15f30dc
code/packages/contracts/contracts/libraries/AccessControl.sol	250cfad
code/packages/contracts/contracts/utils/ReentrancyGuard.sol	608998f
code/packages/contracts/contracts/utils/Volatility.sol	058b0a1
code/packages/contracts/contracts/hedging/UniswapV3HedgingReactor.sol	22305b7
code/packages/contracts/contracts/hedging/PerpHedgingReactor.sol	b09904e
code/packages/contracts/contracts/AlphaPortfolioValuesFeed.sol	0698792
code/packages/contracts/contracts/PortfolioValuesFeed.sol	3c10773
code/packages/contracts/contracts/tokens/ERC20.sol	f3d92d7
code/packages/contracts/contracts/tokens/WETH.sol	01fdd02
code/packages/contracts/contracts/tokens/MintableERC20.sol	3a50497

Contracts Description Table

Contract	Type	Bases
└	**Function Name**	**Visibility**
Accounting	Implementation	IAccounting
└		Public ❗️
└	calculateTokenPrice	Internal 🔒
└	deposit	External ❗️
└	redeem	External ❗️
└	initiateWithdraw	External ❗️
└	completeWithdraw	External ❗️
└	executeEpochCalculation	External ❗️
└	sharesForAmount	Public ❗️
└	amountForShares	Public ❗️
AlphaOptionHandler	Implementation	AccessControl ReentrancyGuar
└		Public ❗️
└	setCustomOrderBounds	External ❗️
└	createOrder	Public ❗️
└	createStrangle	External ❗️
└	executeOrder	Public ❗️
└	executeBuyBackOrder	Public ❗️
└	executeStrangle	External ❗️
└	getOptionRegistry	Internal 🔒
└	getPortfolioValuesFeed	Internal 🔒
└	_getUnderlyingPrice	Internal 🔒
PriceFeed	Implementation	AccessControl
└		Public ❗️
└	addPriceFeed	Public ❗️
└	getRate	External ❗️
└	getNormalizedRate	External ❗️

Contract	Type	Bases
LiquidityPool	Implementation	ERC20, AccessControl ReentrancyGuar Pausable
└		Public ❗
└	pause	External ❗
└	pauseUnpauseTrading	External ❗
└	unpause	External ❗
└	setHedgingReactorAddress	External ❗
└	removeHedgingReactorAddress	External ❗
└	setNewOptionParams	External ❗
└	setBidAskSpread	External ❗
└	setMaxDiscount	External ❗
└	setCollateralCap	External ❗
└	setBufferPercentage	External ❗
└	setRiskFreeRate	External ❗
└	setMaxTimeDeviationThreshold	External ❗
└	setMaxPriceDeviationThreshold	External ❗
└	changeHandler	External ❗
└	setKeeper	External ❗
└	setUtilizationSkewParams	External ❗
└	rebalancePortfolioDelta	External ❗
└	adjustCollateral	External ❗
└	settleVault	External ❗
└	handlerIssue	External ❗
└	handlerWriteOption	External ❗
└	handlerIssueAndWriteOption	External ❗
└	handlerBuybackOption	External ❗
└	resetEphemeralValues	External ❗

Contract	Type	Bases
└	pauseTradingAndRequest	External ❗
└	executeEpochCalculation	External ❗
└	deposit	External ❗
└	redeem	External ❗
└	initiateWithdraw	External ❗
└	completeWithdraw	External ❗
└	_getNormalizedBalance	Internal 🔒
└	getBalance	Public ❗
└	getExternalDelta	Public ❗
└	getPortfolioDelta	Public ❗
└	quotePriceWithUtilizationGreeks	External ❗
└	addUtilizationPremium	Internal 🔒
└	applyDeltaPremium	Internal 🔒
└	getImpliedVolatility	Public ❗
└	getAssets	External ❗
└	getNAV	External ❗
└	_redeem	Internal 🔒
└	_getNAV	Internal 🔒
└	_getAssets	Internal 🔒
└	_getLiabilities	Internal 🔒
└	checkBuffer	Public ❗
└	_issue	Internal 🔒
└	_writeOption	Internal 🔒
└	_buybackOption	Internal 🔒
└	_adjustVariables	Internal 🔒
└	_getVolatilityFeed	Internal 🔒

Contract	Type	Bases
└	_getPortfolioValuesFeed	Internal 🔒
└	_getAccounting	Internal 🔒
└	_getOptionRegistry	Internal 🔒
└	_getUnderlyingPrice	Internal 🔒
└	_isTradingNotPaused	Internal 🔒
└	_isHandler	Internal 🔒
└	_isKeeper	Internal 🔒
OptionRegistry	Implementation	AccessControl
└		Public ❗
└	setLiquidityPool	External ❗
└	setKeeper	External ❗
└	setHealthThresholds	External ❗
└	issue	External ❗
└	open	External ❗
└	close	External ❗
└	settle	External ❗
└	adjustCollateral	External ❗
└	adjustCollateralCaller	External ❗
└	wCollatLiquidatedVault	External ❗
└	registerLiquidatedVault	External ❗
└	redeem	External ❗
└	getCollateral	External ❗
└	getOtoken	External ❗
└	checkVaultHealth	Public ❗
└	getSeriesAddress	External ❗
└	getSeries	External ❗
└	getSeriesInfo	External ❗
└	getIssuanceHash	Public ❗

Contract	Type	Bases
└	getIssuanceHash	Internal 🔒
└	formatStrikePrice	Public ❗
└	_isLiquidityPool	Internal 🔒
└	_isKeeper	Internal 🔒
OptionHandler	Implementation	Pausable, AccessControl ReentrancyGuar
└		Public ❗
└	setCustomOrderBounds	External ❗
└	pause	External ❗
└	unpause	External ❗
└	addOrRemoveBuybackAddress	External ❗
└	setMinDeltaForRequest	External ❗
└	createOrder	Public ❗
└	createStrangle	External ❗
└	executeOrder	Public ❗
└	executeStrangle	External ❗
└	issueAndWriteOption	External ❗
└	issue	External ❗
└	writeOption	External ❗
└	buybackOption	External ❗
└	getOptionRegistry	Internal 🔒
└	getPortfolioValuesFeed	Internal 🔒
└	_getUnderlyingPrice	Internal 🔒
Protocol	Implementation	AccessControl

Contract	Type	Bases
└		Public ❗
└	changeVolatilityFeed	External ❗
└	changePortfolioValuesFeed	External ❗
└	changeAccounting	External ❗
└	changePriceFeed	External ❗
Authority	Implementation	IAuthority, AccessControl
└		Public ❗
└	pushGovernor	External ❗
└	pushGuardian	External ❗
└	pushManager	External ❗
└	pullGovernor	External ❗
└	revokeGuardian	External ❗
└	pullManager	External ❗
CustomErrors	Interface	
BlackScholes	Library	
└	callOptionPrice	Public ❗
└	callOptionPriceGreeks	Public ❗
└	putOptionPriceGreeks	Public ❗
└	putOptionPrice	Public ❗
└	getTimeStamp	Private 🔐
└	getD1	Private 🔐
└	getIntermediates	Private 🔐
└	blackScholesCalc	Public ❗
└	blackScholesCalcGreeks	Public ❗
└	getDelta	Public ❗
OptionsCompute	Library	
└	convertToDecimals	Internal 🔒

Contract	Type	Bases
└	convertFromDecimals	Internal 🔒
└	convertToCollateralDenominated	Internal 🔒
└	calculatePercentageChange	Internal 🔒
└	validatePortfolioValues	Public ❗️
└	getUtilizationPrice	Internal 🔒
└	quotePriceGreeks	Internal 🔒
EnumerableSet	Library	
└	_add	Private 🔐
└	_remove	Private 🔐
└	_contains	Private 🔐
└	_length	Private 🔐
└	_at	Private 🔐
└	_values	Private 🔐
└	add	Internal 🔒
└	remove	Internal 🔒
└	contains	Internal 🔒
└	length	Internal 🔒
└	at	Internal 🔒
└	values	Internal 🔒
NormalDist	Library	
└	cdf	Public ❗️
└	phi	Public ❗️
└	getScoresFromT	Public ❗️
SafeTransferLib	Library	
└	safeTransferETH	Internal 🔒
└	safeTransferFrom	Internal 🔒
└	safeTransfer	Internal 🔒
└	safeApprove	Internal 🔒

Contract	Type	Bases
└	didLastOptionalReturnCallSucceed	Private 🔐
OpynInteractions	Library	
└	getOrDeployOtoken	External ❗️
└	getOtoken	External ❗️
└	createShort	External ❗️
└	depositCollat	External ❗️
└	withdrawCollat	External ❗️
└	burnShort	External ❗️
└	settle	External ❗️
└	redeem	External ❗️
Types	Library	
AccessControl	Implementation	
└		Public ❗️
└	setAuthority	External ❗️
└	_onlyGovernor	Internal 🔒
└	_onlyGuardian	Internal 🔒
└	_onlyManager	Internal 🔒
ReentrancyGuard	Implementation	
└		Public ❗️
Volatility	Implementation	
└	computeIVFromSkewInts	Public ❗️
└	computeIVFromSkew	Internal 🔒
UniswapV3HedgingReactor	Implementation	IHedgingReacto AccessControl
└		Public ❗️
└	changePoolFee	External ❗️
└	setMinAmount	External ❗️

Contract	Type	Bases
└	setSlippage	External ❗
└	hedgeDelta	External ❗
└	withdraw	External ❗
└	update	External ❗
└	getDelta	External ❗
└	getPoolDenominatedValue	External ❗
└	_swapExactOutputSingle	Internal 🔒
└	_swapExactInputSingle	Internal 🔒
└	getUnderlyingPrice	Internal 🔒
PerpHedgingReactor	Implementation	IHedgingReacto AccessControl
└		Public ❗
└	setHealthFactor	External ❗
└	setKeeper	External ❗
└	setSyncOnChange	External ❗
└	initialiseReactor	External ❗
└	hedgeDelta	External ❗
└	withdraw	External ❗
└	syncAndUpdate	External ❗
└	sync	Public ❗
└	update	Public ❗
└	getDelta	External ❗
└	getPoolDenominatedValue	External ❗
└	checkVaultHealth	External ❗
└	_changePosition	Internal 🔒
└	_isKeeper	Internal 🔒
AlphaPortfolioValuesFeed	Implementation	AccessControl IPortfolioValuesFe

Contract	Type	Bases
└		Public ❗
└	setLiquidityPool	External ❗
└	setProtocol	External ❗
└	setRFR	External ❗
└	setKeeper	External ❗
└	setHandler	External ❗
└	fulfill	External ❗
└	updateStores	External ❗
└	syncLooper	External ❗
└	cleanLooperManually	External ❗
└	_cleanLooper	Internal 🔒
└	accountLiquidatedSeries	External ❗
└	migrate	External ❗
└	requestPortfolioData	External ❗
└	getPortfolioValues	External ❗
└	_isKeeper	Internal 🔒
└	_isHandler	Internal 🔒
└	isAddressInSet	External ❗
└	addressAtIndexInSet	External ❗
└	addressSetLength	External ❗
└	getAddressSet	External ❗
└	_getVolatilityFeed	Internal 🔒
└	_getOptionRegistry	Internal 🔒
└	_getUnderlyingPrice	Internal 🔒
PortfolioValuesFeed	Implementation	AccessControl ChainlinkClient
└		Public ❗
└	setOracle	External ❗
└	setLiquidityPool	External ❗

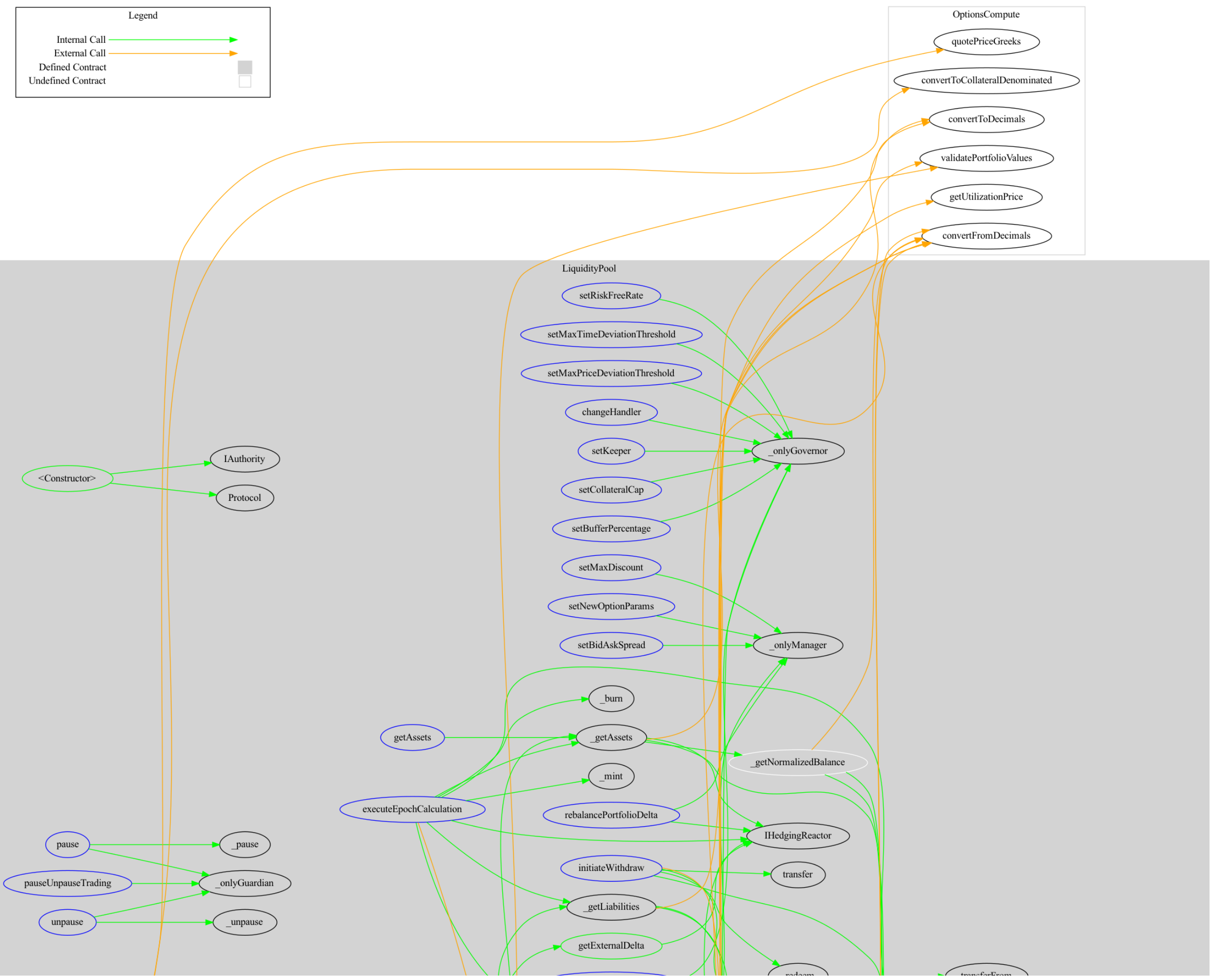
Contract	Type	Bases
└	setAddressStringMapping	External ❗
└	setLink	External ❗
└	setKeeper	External ❗
└	fulfill	External ❗
└	withdrawLink	External ❗
└	requestPortfolioData	External ❗
└	getPortfolioValues	External ❗
└	_isKeeper	Internal 🔒
ERC20	Implementation	
└		Public ❗
└	approve	Public ❗
└	transfer	Public ❗
└	transferFrom	Public ❗
└	permit	Public ❗
└	DOMAIN_SEPARATOR	Public ❗
└	computeDomainSeparator	Internal 🔒
└	_mint	Internal 🔒
└	_burn	Internal 🔒
WETH	Implementation	
└	deposit	Public ❗
└	withdraw	Public ❗
└	totalSupply	Public ❗
└	approve	Public ❗
└	transfer	Public ❗
└	transferFrom	Public ❗
MintableERC20	Implementation	
└		Public ❗
└	approve	Public ❗

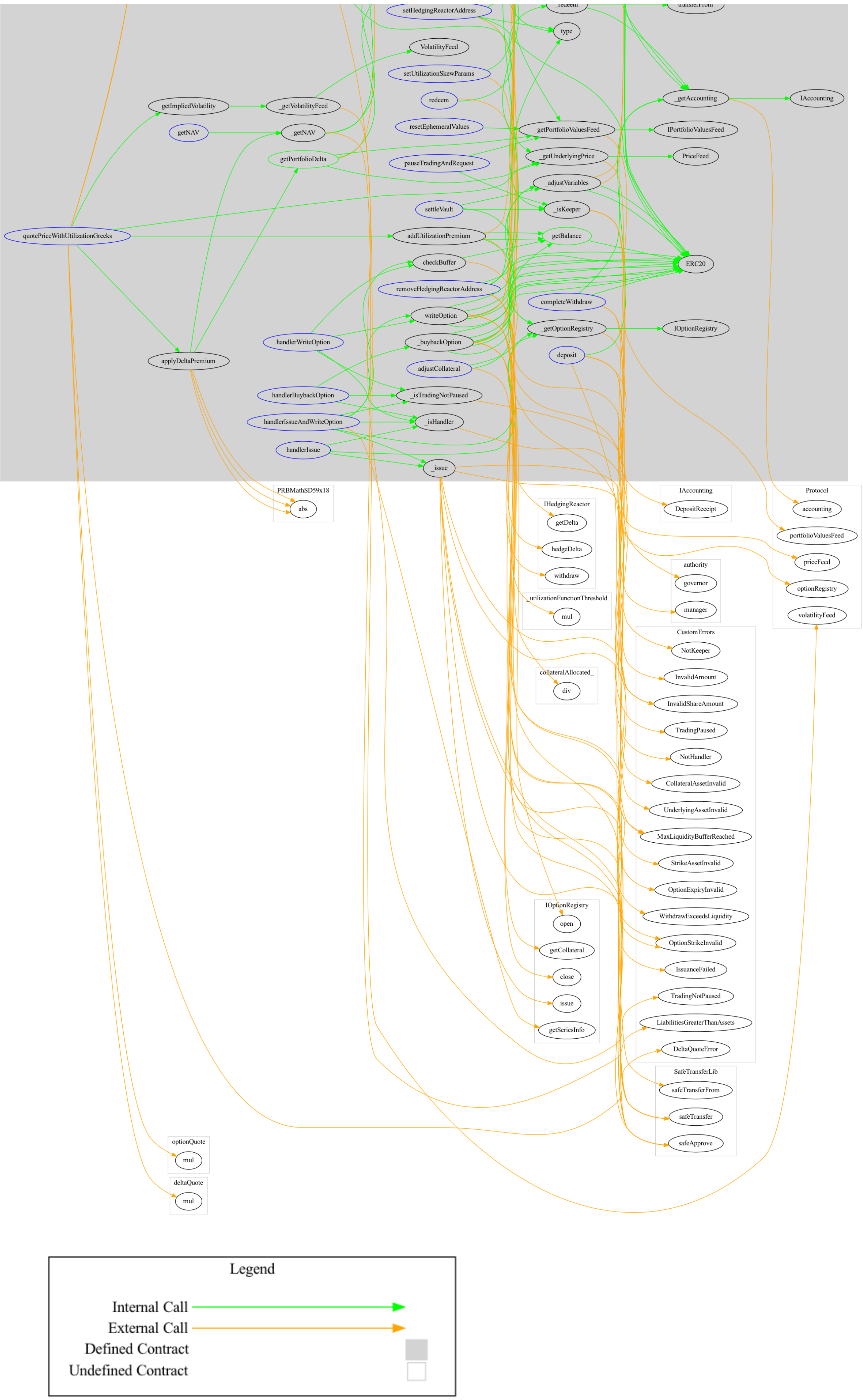
Contract	Type	Bases
└	transfer	Public ❗️
└	transferFrom	Public ❗️
└	permit	Public ❗️
└	DOMAIN_SEPARATOR	Public ❗️
└	computeDomainSeparator	Internal 🔒
└	mint	External ❗️
└	_mint	Internal 🔒
└	_burn	Internal 🔒

Legend

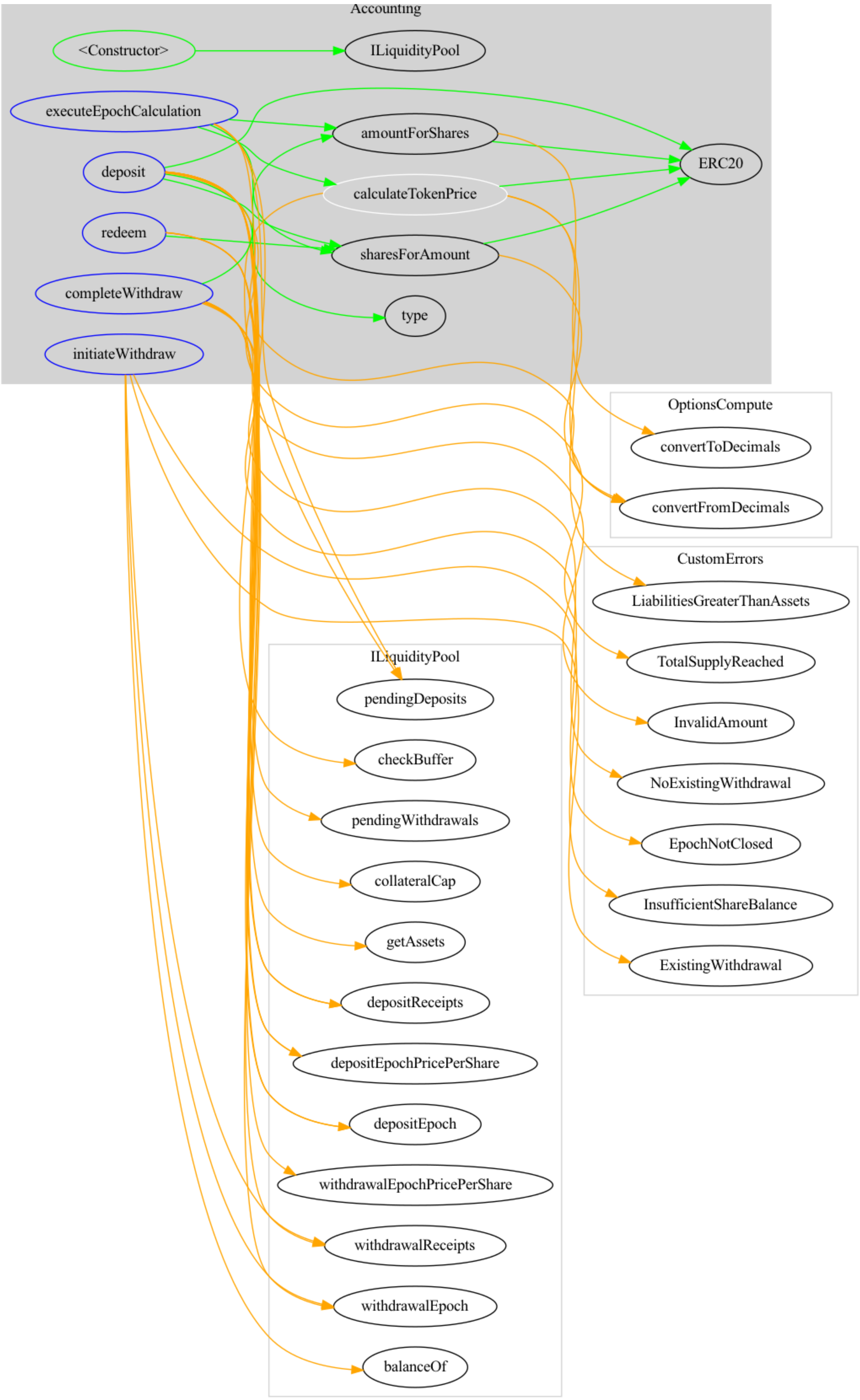
Symbol	Meaning
🛑	Function can modify state
💵	Function is payable

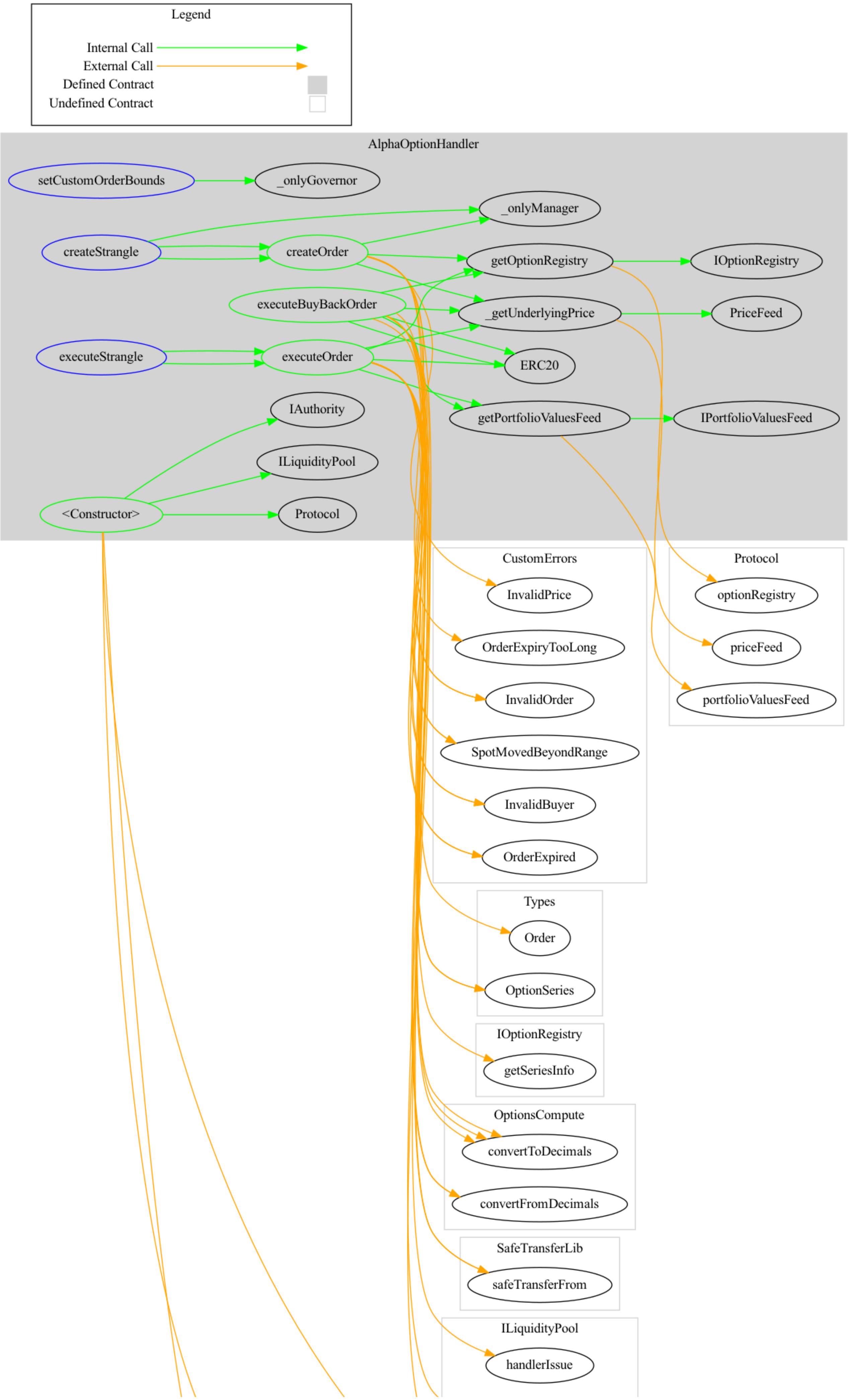
Graphs

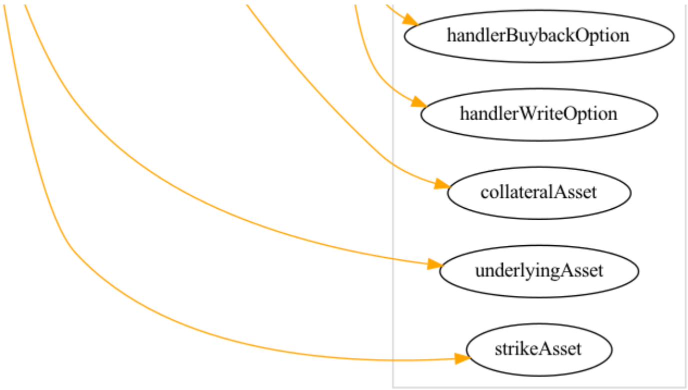


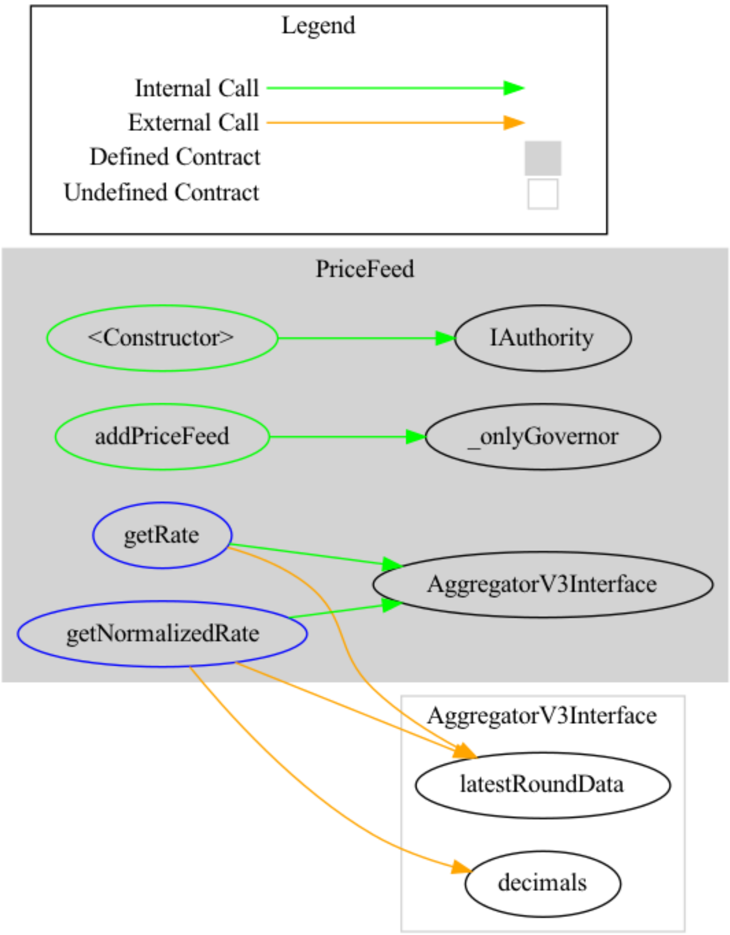


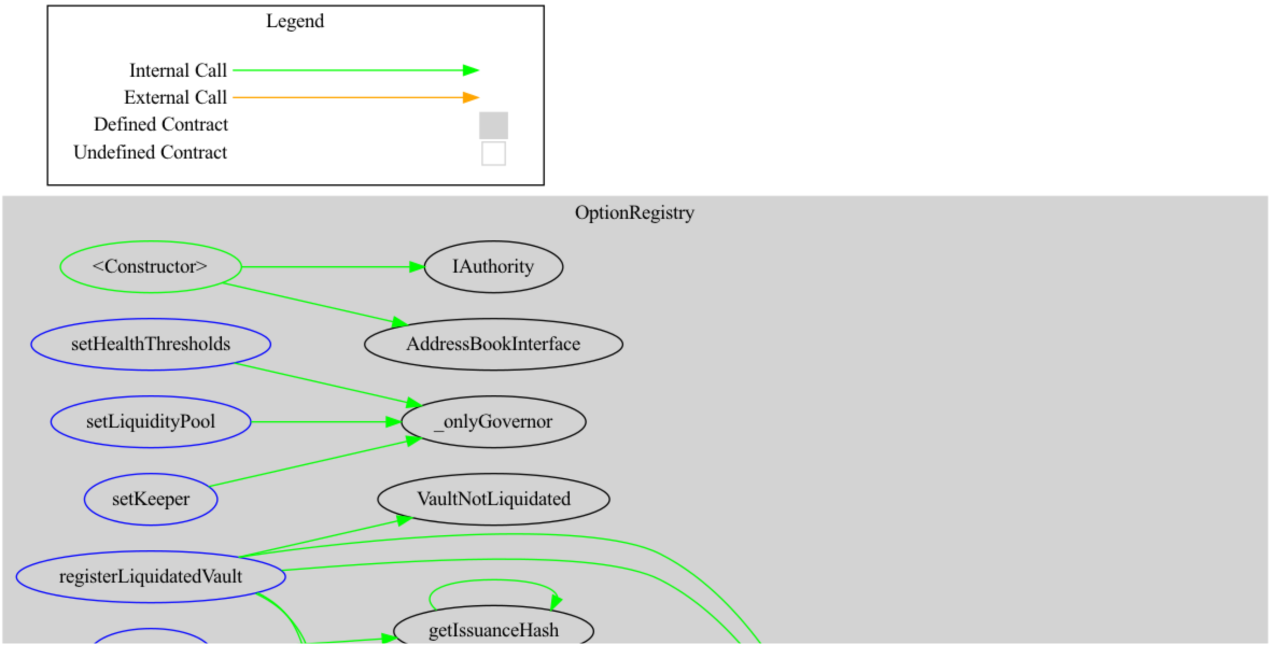
setHedgingReactorAddress
_redeem
transferFrom
type
VolatilityFeed
setUtilizationSkewParams
redeem
_getAccounting
IAccounting
getImpliedVolatility
_getVolatilityFeed
getNAV
_getNAV
resetEphemeralValues
_getPortfolioValuesFeed
IPortfolioValuesFeed
getPortfolioDelta
pauseTradingAndRequest
_getUnderlyingPrice
PriceFeed
_adjustVariables
settleVault
_isKeeper
quotePriceWithUtilizationGreeks
addUtilizationPremium
getBalance
checkBuffer
ERC20
removeHedgingReactorAddress
completeWithdraw
_writeOption
handlerWriteOption
_getOptionRegistry
IOptionRegistry
_buybackOption
applyDeltaPremium
deposit
adjustCollateral
handlerBuybackOption
_isTradingNotPaused
handlerIssueAndWriteOption
_isHandler
handlerIssue
_issue
PRBMathSD59x18
abs
IHedgingReactor
getDelta
hedgeDelta
withdraw
IAccounting
DepositReceipt
Protocol
accounting
portfolioValuesFeed
priceFeed
optionRegistry
volatilityFeed
authority
governor
manager
_utilizationFunctionThreshold
mul
collateralAllocated_
div
CustomErrors
NotKeeper
InvalidAmount
InvalidShareAmount
TradingPaused
NotHandler
CollateralAssetInvalid
UnderlyingAssetInvalid
MaxLiquidityBufferReached
StrikeAssetInvalid
OptionExpiryInvalid
WithdrawExceedsLiquidity
OptionStrikeInvalid
IssuanceFailed
TradingNotPaused
LiabilitiesGreaterThanAssets
DeltaQuoteError
IOptionRegistry
open
getCollateral
close
issue
getSeriesInfo
SafeTransferLib
safeTransferFrom
safeTransfer
safeApprove
optionQuote
mul
deltaQuote
mul
Legend
Internal Call
External Call
Defined Contract
Undefined Contract

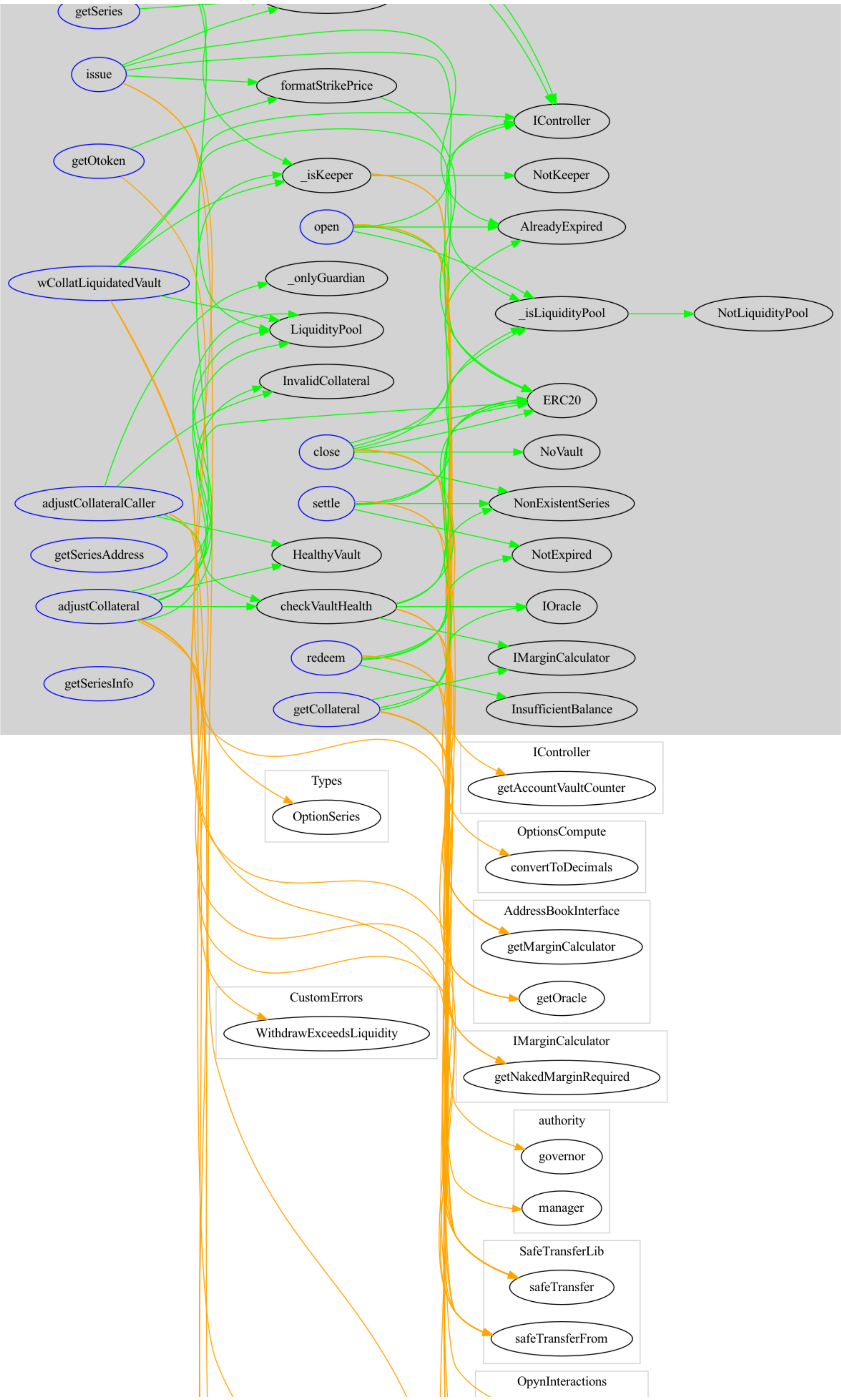

Accounting
<Constructor>
ILiquidityPool
executeEpochCalculation
amountForShares
deposit
ERC20
calculateTokenPrice
redeem
sharesForAmount
completeWithdraw
type
initiateWithdraw
OptionsCompute
convertToDecimals
convertFromDecimals
CustomErrors
LiabilitiesGreaterThanAssets
TotalSupplyReached
ILiquidityPool
pendingDeposits
InvalidAmount
checkBuffer
NoExistingWithdrawal
pendingWithdrawals
EpochNotClosed
collateralCap
InsufficientShareBalance
getAssets
ExistingWithdrawal
depositReceipts
depositEpochPricePerShare
depositEpoch
withdrawalEpochPricePerShare
withdrawalReceipts
withdrawalEpoch
balanceOf

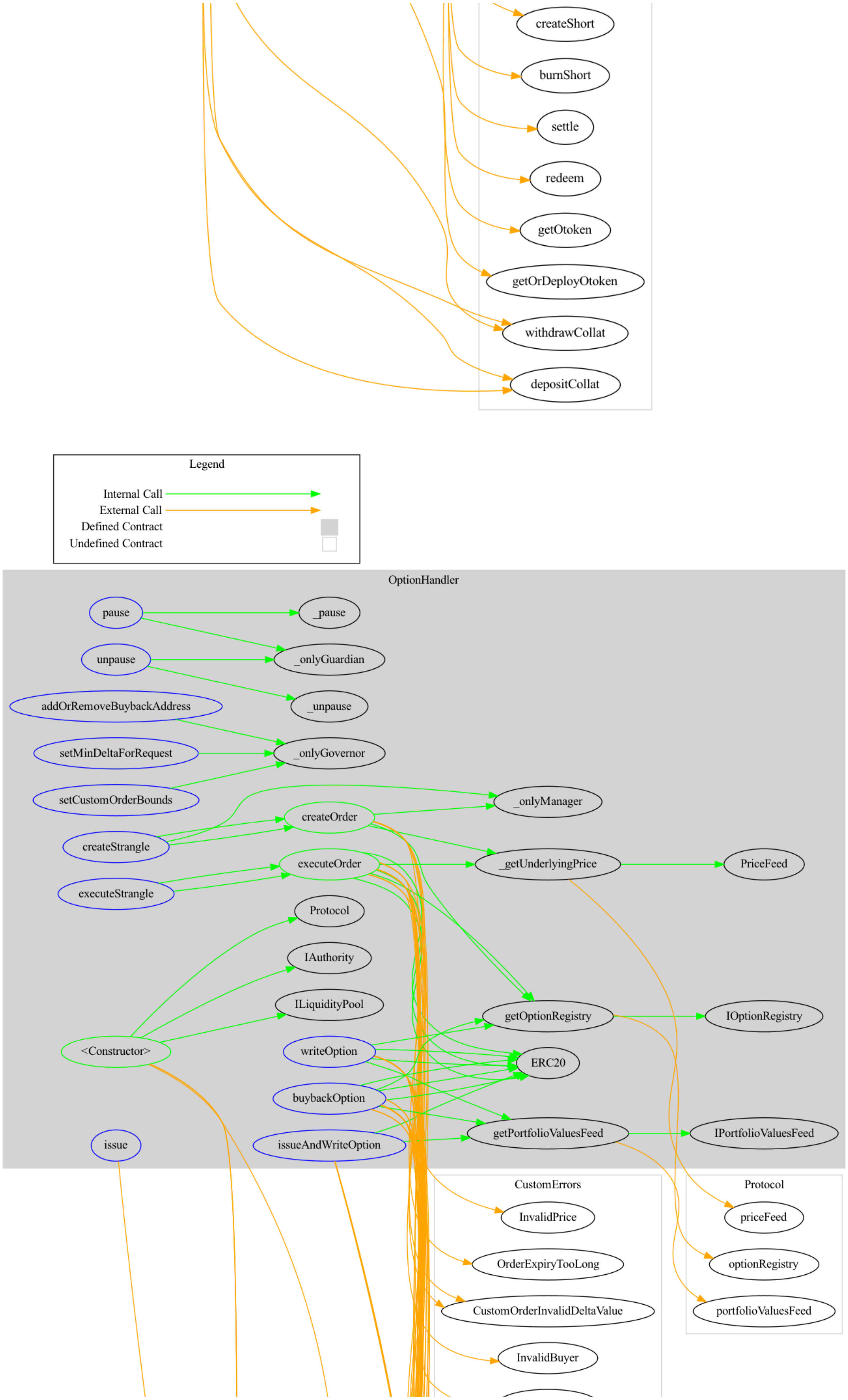

Legend
Internal Call
External Call
Defined Contract
Undefined Contract
AlphaOptionHandler
setCustomOrderBounds
_onlyGovernor
_onlyManager
createStrangle
createOrder
getOptionRegistry
IOptionRegistry
executeBuyBackOrder
_getUnderlyingPrice
PriceFeed
executeStrangle
executeOrder
ERC20
getPortfolioValuesFeed
IPortfolioValuesFeed
IAuthority
ILiquidityPool
<Constructor>
Protocol
CustomErrors
InvalidPrice
OrderExpiryTooLong
InvalidOrder
SpotMovedBeyondRange
InvalidBuyer
OrderExpired
Protocol
optionRegistry
priceFeed
portfolioValuesFeed
Types
Order
OptionSeries
IOptionRegistry
getSeriesInfo
OptionsCompute
convertToDecimals
convertFromDecimals
SafeTransferLib
safeTransferFrom
ILiquidityPool
handlerIssue

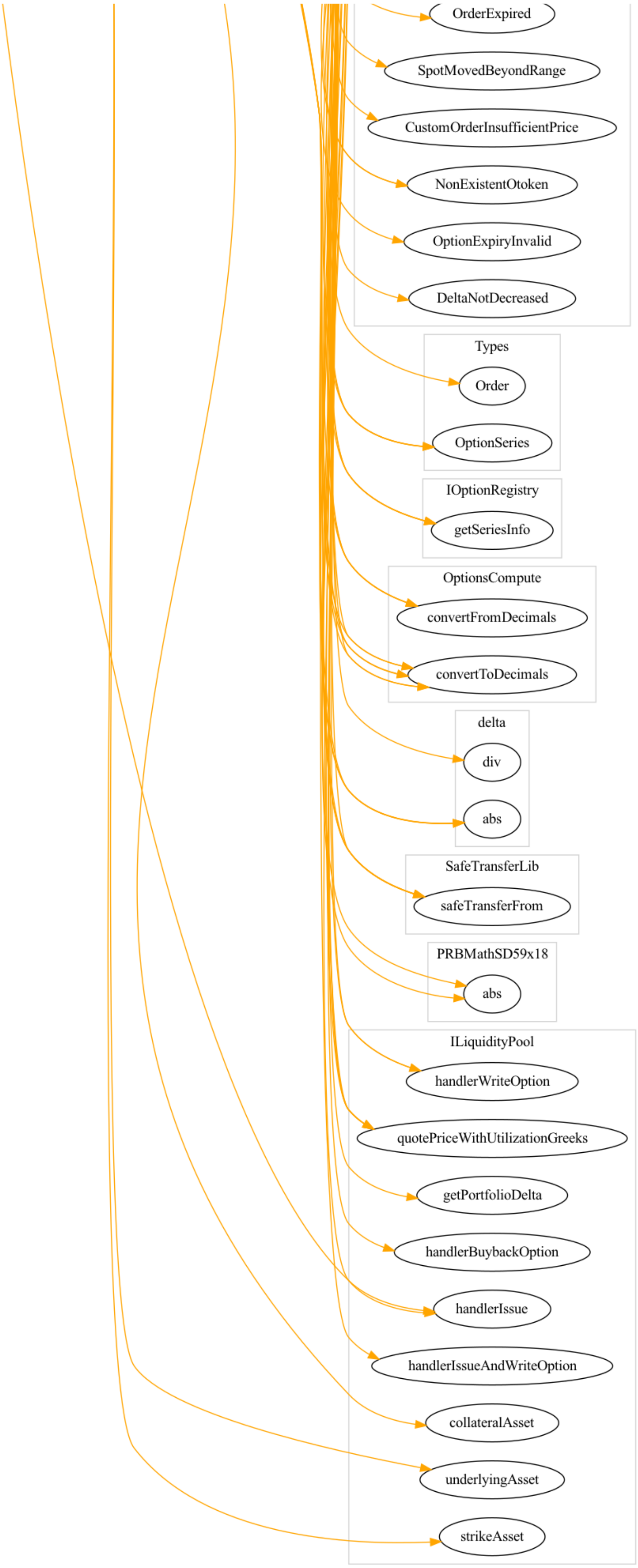

handlerBuybackOption
handlerWriteOption
collateralAsset
underlyingAsset
strikeAsset

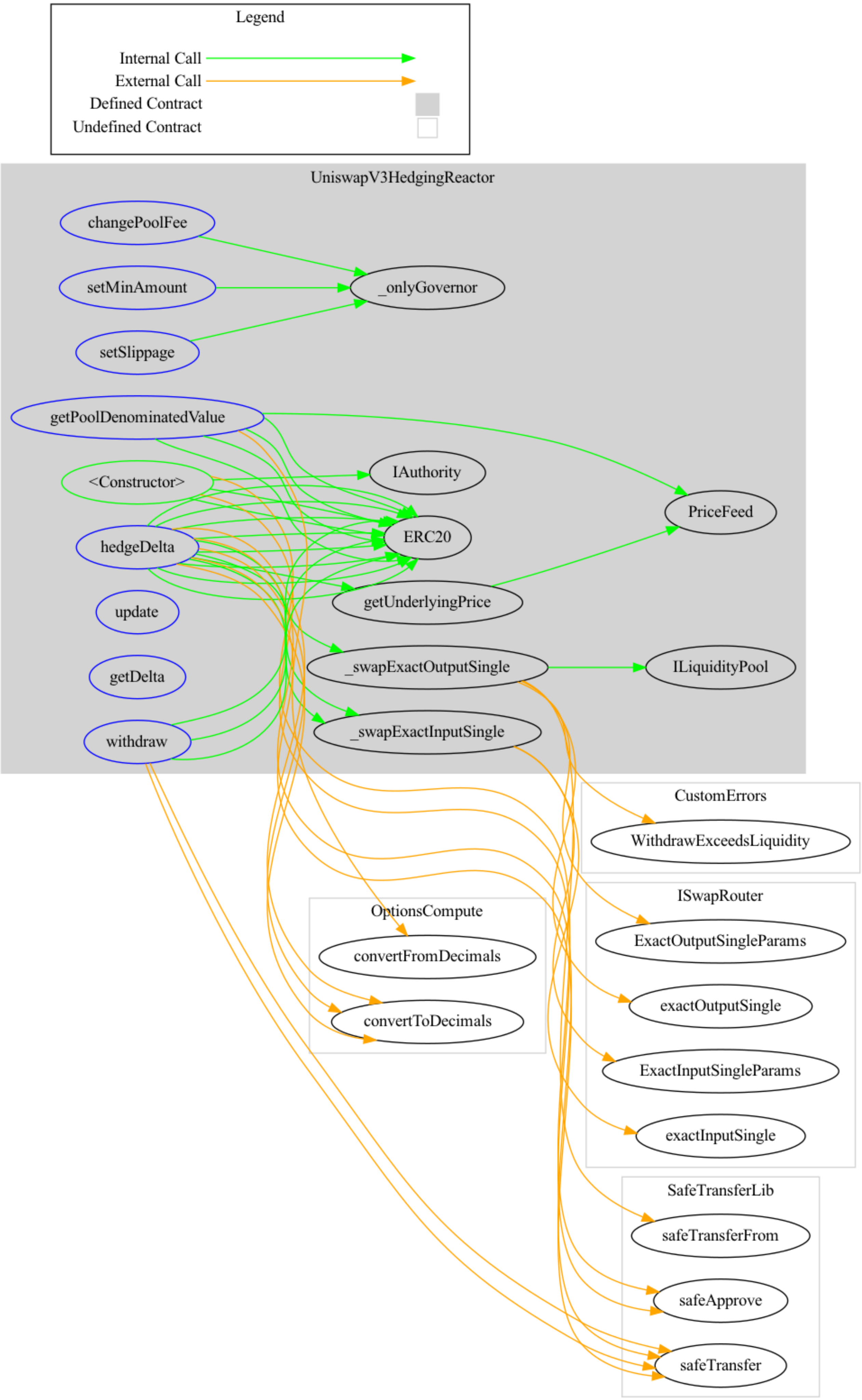

Legend
Internal Call
External Call
Defined Contract
Undefined Contract
PriceFeed
<Constructor>
IAuthority
addPriceFeed
_onlyGovernor
getRate
AggregatorV3Interface
getNormalizedRate
AggregatorV3Interface
latestRoundData
decimals

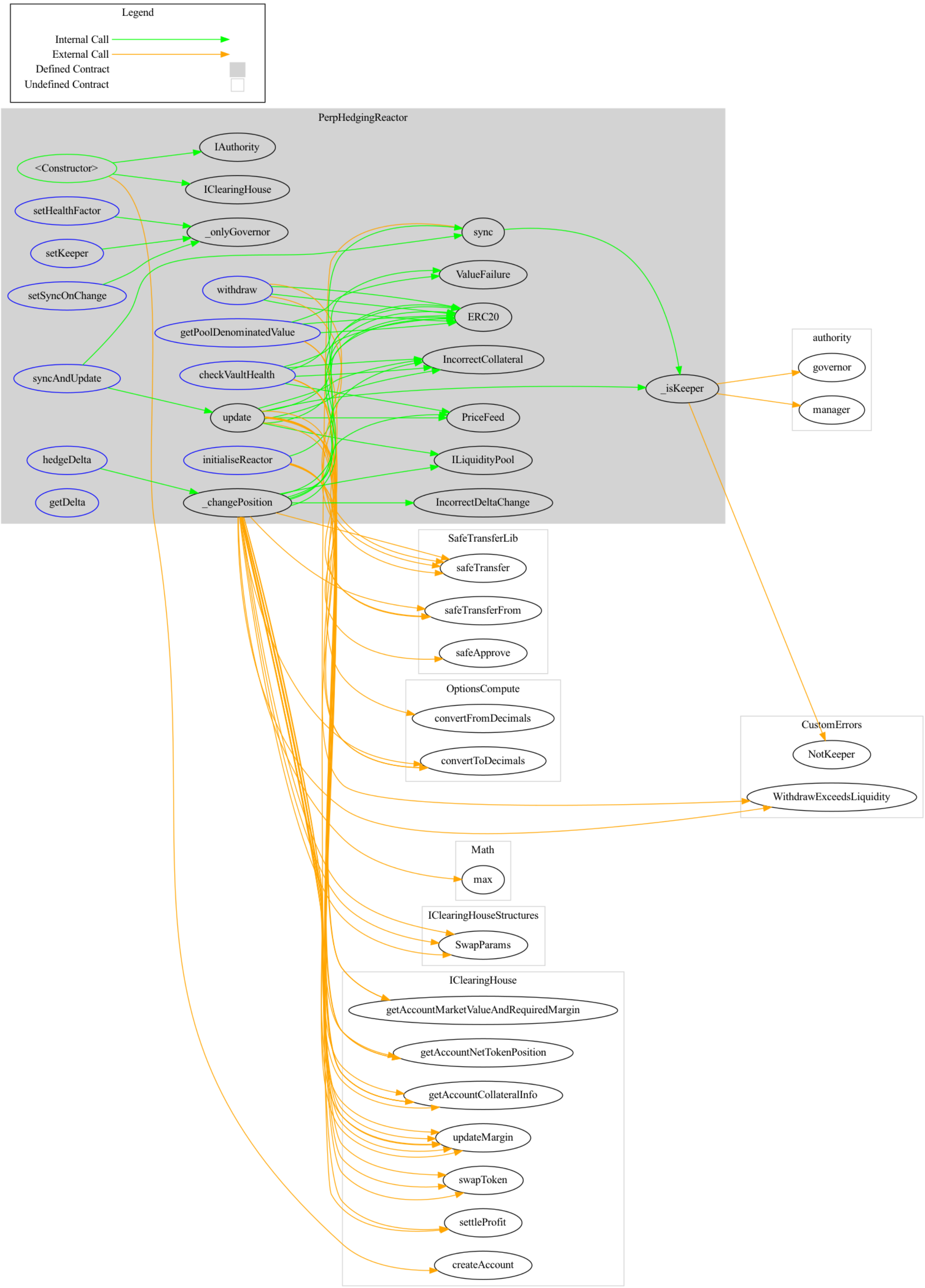

Legend
Internal Call
External Call
Defined Contract
Undefined Contract
OptionRegistry
<Constructor>
IAuthority
setHealthThresholds
AddressBookInterface
setLiquidityPool
_onlyGovernor
setKeeper
VaultNotLiquidated
registerLiquidatedVault
getIssuanceHash

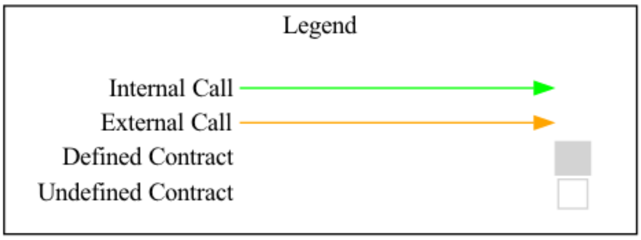

getSeries
issue
formatStrikePrice
IController
getOtoken
_isKeeper
NotKeeper
open
AlreadyExpired
wCollatLiquidatedVault
_onlyGuardian
_isLiquidityPool
NotLiquidityPool
LiquidityPool
InvalidCollateral
ERC20
close
NoVault
adjustCollateralCaller
settle
NonExistentSeries
getSeriesAddress
HealthyVault
NotExpired
adjustCollateral
checkVaultHealth
IOracle
redeem
IMarginCalculator
getSeriesInfo
getCollateral
InsufficientBalance
IController
getAccountVaultCounter
Types
OptionSeries
OptionsCompute
convertToDecimals
AddressBookInterface
getMarginCalculator
getOracle
CustomErrors
WithdrawExceedsLiquidity
IMarginCalculator
getNakedMarginRequired
authority
governor
manager
SafeTransferLib
safeTransfer
safeTransferFrom
OpynInteractions

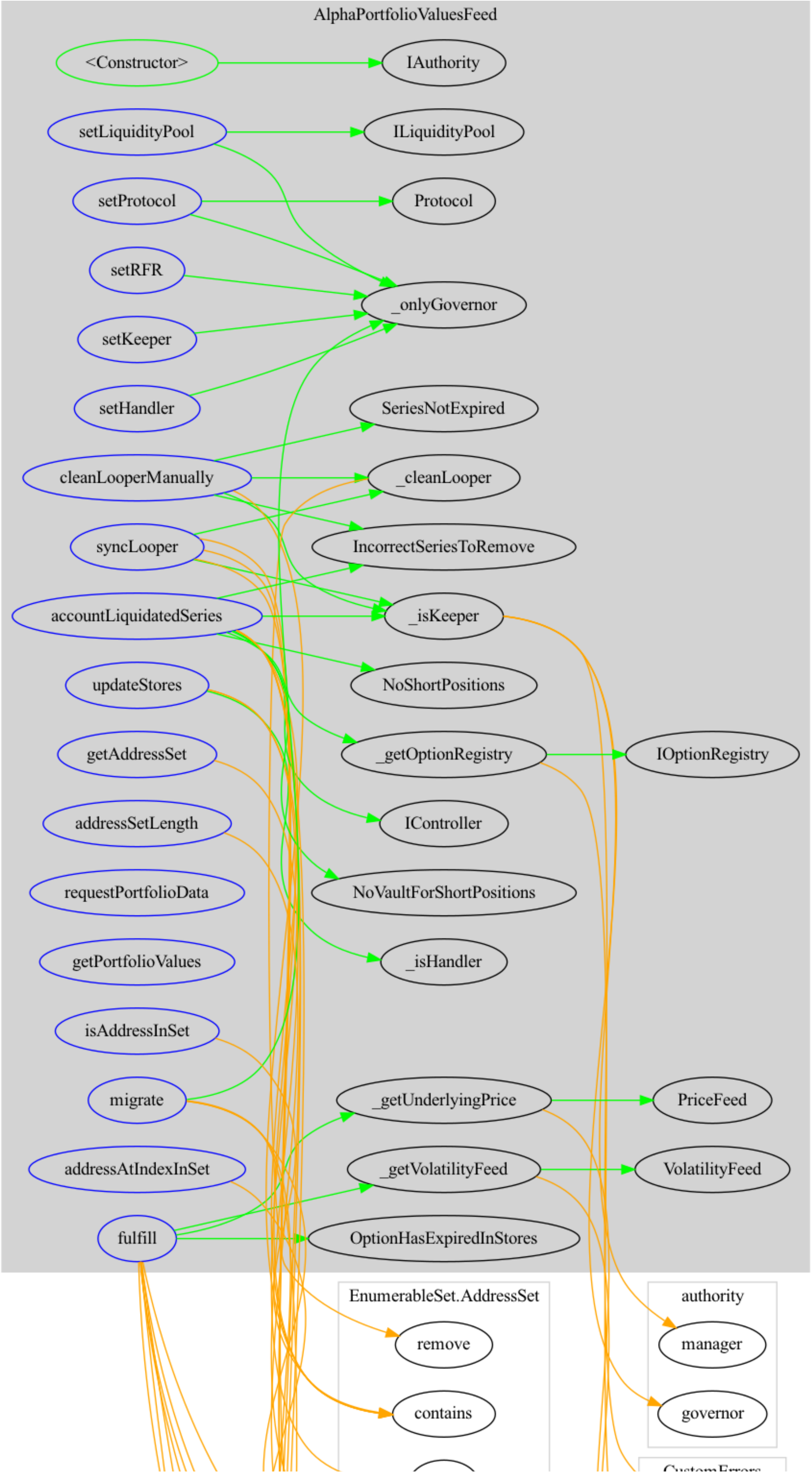

createShort
burnShort
settle
redeem
getOtoken
getOrDeployOtoken
withdrawCollat
depositCollat
Legend
Internal Call
External Call
Defined Contract
Undefined Contract
OptionHandler
pause
_pause
unpause
_onlyGuardian
_unpause
addOrRemoveBuybackAddress
setMinDeltaForRequest
_onlyGovernor
setCustomOrderBounds
_onlyManager
createOrder
createStrangle
executeOrder
_getUnderlyingPrice
PriceFeed
executeStrangle
Protocol
IAuthority
ILiquidityPool
getOptionRegistry
IOptionRegistry
<Constructor>
writeOption
ERC20
buybackOption
issue
issueAndWriteOption
getPortfolioValuesFeed
IPortfolioValuesFeed
CustomErrors
InvalidPrice
OrderExpiryTooLong
CustomOrderInvalidDeltaValue
InvalidBuyer
Protocol
priceFeed
optionRegistry
portfolioValuesFeed

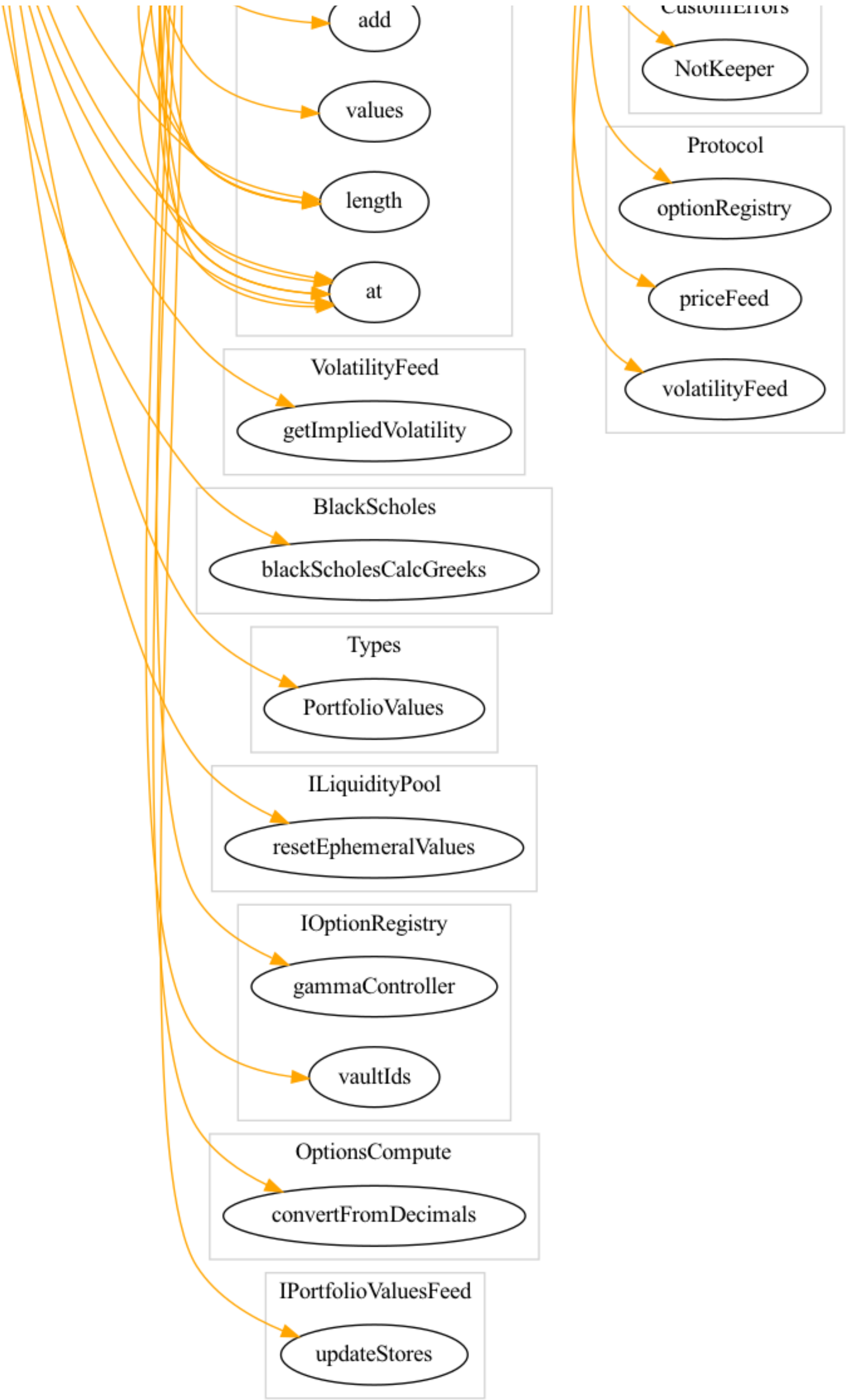

OrderExpired
SpotMovedBeyondRange
CustomOrderInsufficientPrice
NonExistentOtoken
OptionExpiryInvalid
DeltaNotDecreased
Types
Order
OptionSeries
IOptionRegistry
getSeriesInfo
OptionsCompute
convertFromDecimals
convertToDecimals
delta
div
abs
SafeTransferLib
safeTransferFrom
PRBMathSD59x18
abs
ILiquidityPool
handlerWriteOption
quotePriceWithUtilizationGreeks
getPortfolioDelta
handlerBuybackOption
handlerIssue
handlerIssueAndWriteOption
collateralAsset
underlyingAsset
strikeAsset

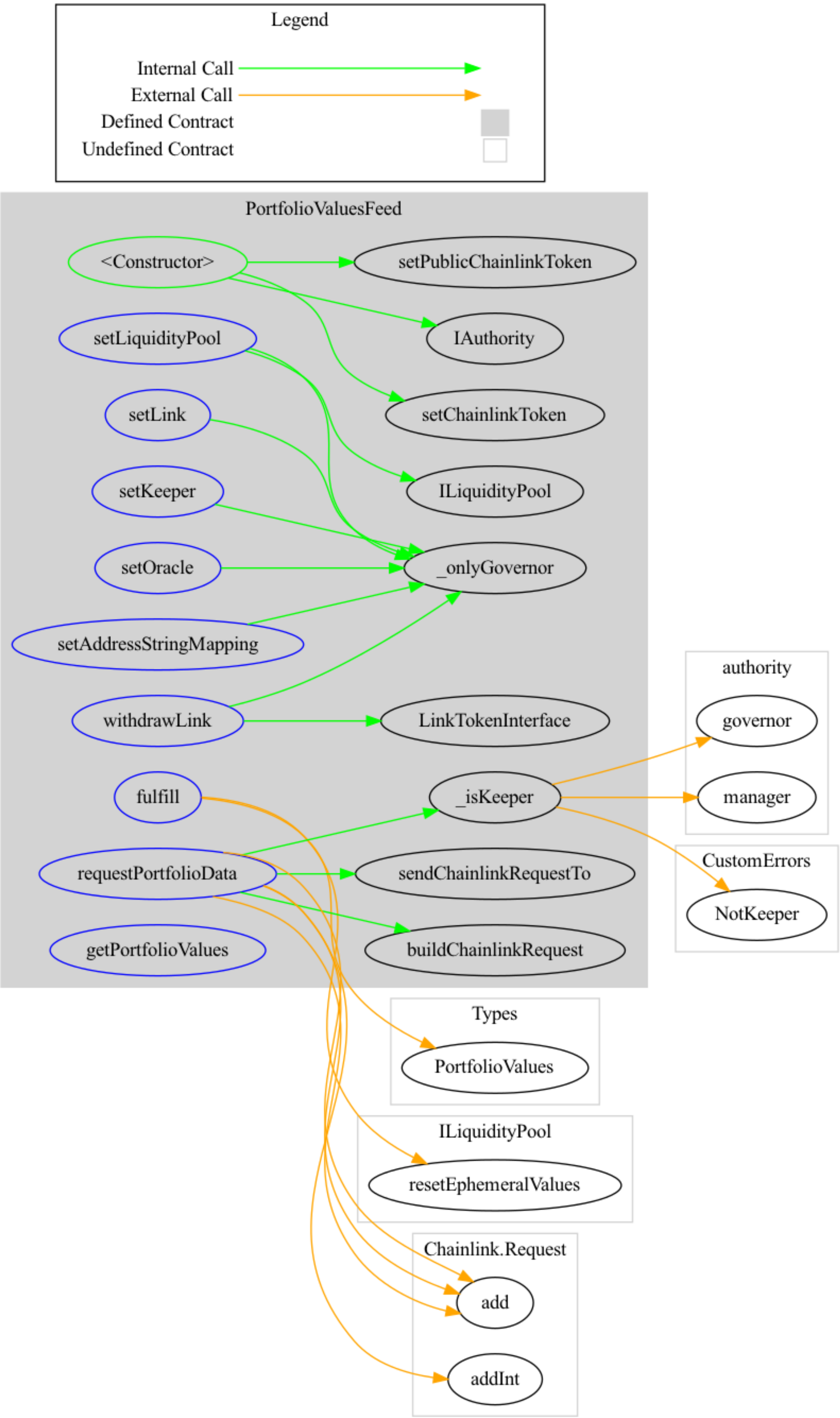

Legend
Internal Call
External Call
Defined Contract
Undefined Contract
UniswapV3HedgingReactor
changePoolFee
setMinAmount
setSlippage
_onlyGovernor
getPoolDenominatedValue
<Constructor>
IAuthority
ERC20
PriceFeed
hedgeDelta
update
getDelta
withdraw
getUnderlyingPrice
_swapExactOutputSingle
ILiquidityPool
_swapExactInputSingle
CustomErrors
WithdrawExceedsLiquidity
ISwapRouter
ExactOutputSingleParams
exactOutputSingle
ExactInputSingleParams
exactInputSingle
OptionsCompute
convertFromDecimals
convertToDecimals
SafeTransferLib
safeTransferFrom
safeApprove
safeTransfer


Legend
Internal Call
External Call
Defined Contract
Undefined Contract
PerpHedgingReactor
<Constructor>
IAuthority
IClearingHouse
setHealthFactor
_onlyGovernor
setKeeper
setSyncOnChange
sync
withdraw
ValueFailure
getPoolDenominatedValue
ERC20
checkVaultHealth
IncorrectCollateral
syncAndUpdate
_isKeeper
update
PriceFeed
hedgeDelta
initialiseReactor
ILiquidityPool
getDelta
_changePosition
IncorrectDeltaChange
authority
governor
manager
SafeTransferLib
safeTransfer
safeTransferFrom
safeApprove
OptionsCompute
convertFromDecimals
convertToDecimals
CustomErrors
NotKeeper
WithdrawExceedsLiquidity
Math
max
IClearingHouseStructures
SwapParams
IClearingHouse
getAccountMarketValueAndRequiredMargin
getAccountNetTokenPosition
getAccountCollateralInfo
updateMargin
swapToken
settleProfit
createAccount

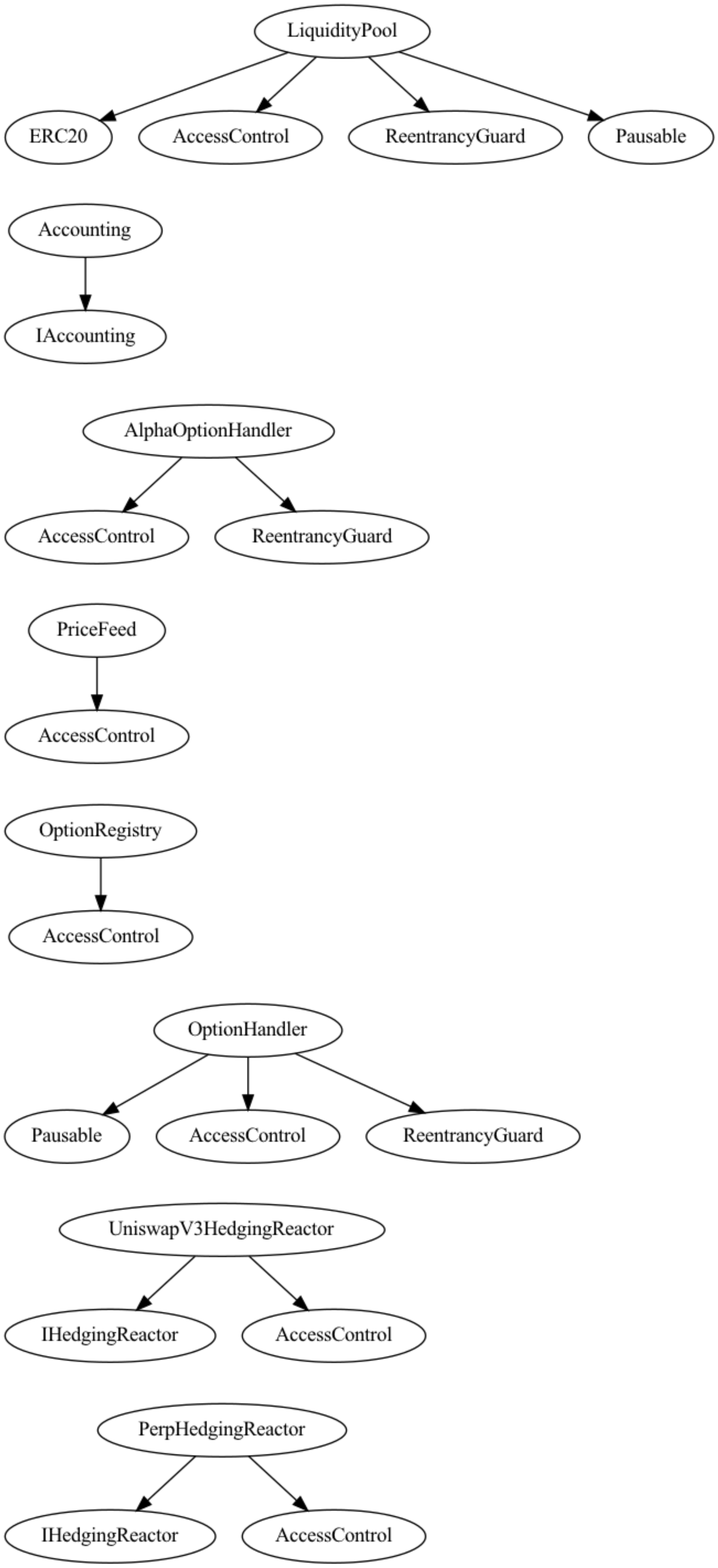

Legend
Internal Call
External Call
Defined Contract
Undefined Contract

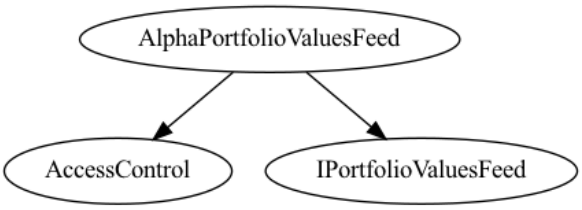

AlphaPortfolioValuesFeed
<Constructor>
IAuthority
setLiquidityPool
ILiquidityPool
setProtocol
Protocol
setRFR
_onlyGovernor
setKeeper
setHandler
SeriesNotExpired
cleanLooperManually
_cleanLooper
syncLooper
IncorrectSeriesToRemove
accountLiquidatedSeries
_isKeeper
updateStores
NoShortPositions
getAddressSet
_getOptionRegistry
IOptionRegistry
addressSetLength
IController
requestPortfolioData
NoVaultForShortPositions
getPortfolioValues
_isHandler
isAddressInSet
migrate
_getUnderlyingPrice
PriceFeed
addressAtIndexInSet
_getVolatilityFeed
VolatilityFeed
fulfill
OptionHasExpiredInStores
EnumerableSet.AddressSet
remove
contains
authority
manager
governor
CustomErrors

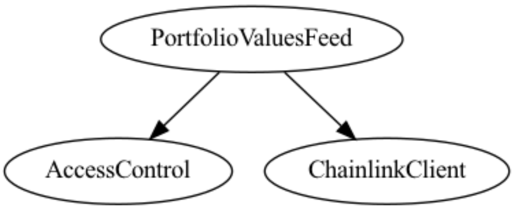

add
values
length
at
CustomErrors
NotKeeper
Protocol
optionRegistry
priceFeed
volatilityFeed
VolatilityFeed
getImpliedVolatility
BlackScholes
blackScholesCalcGreeks
Types
PortfolioValues
ILiquidityPool
resetEphemeralValues
IOptionRegistry
gammaController
vaultIds
OptionsCompute
convertFromDecimals
IPortfolioValuesFeed
updateStores



Inheritance


LiquidityPool
ERC20
AccessControl
ReentrancyGuard
Pausable
Accounting
IAccounting
AlphaOptionHandler
AccessControl
ReentrancyGuard
PriceFeed
AccessControl
OptionRegistry
AccessControl
OptionHandler
Pausable
AccessControl
ReentrancyGuard
UniswapV3HedgingReactor
IHedgingReactor
AccessControl
PerpHedgingReactor
IHedgingReactor
AccessControl





Describe

```
$ npx surya describe code/packages/contracts/contracts/LiquidityPool.sol code/packages/contracts/contracts

 +  LiquidityPool (ERC20, AccessControl, ReentrancyGuard, Pausable)
    - [Pub] <Constructor> #
       - modifiers: ERC20,AccessControl
    - [Ext] pause #
    - [Ext] pauseUnpauseTrading #
    - [Ext] unpause #
    - [Ext] setHedgingReactorAddress #
    - [Ext] removeHedgingReactorAddress #
    - [Ext] setNewOptionParams #
    - [Ext] setBidAskSpread #
    - [Ext] setMaxDiscount #
    - [Ext] setCollateralCap #
    - [Ext] setBufferPercentage #
    - [Ext] setRiskFreeRate #
    - [Ext] setMaxTimeDeviationThreshold #
    - [Ext] setMaxPriceDeviationThreshold #
    - [Ext] changeHandler #
    - [Ext] setKeeper #
    - [Ext] setUtilizationSkewParams #
    - [Ext] rebalancePortfolioDelta #
    - [Ext] adjustCollateral #
    - [Ext] settleVault #
    - [Ext] handlerIssue #
    - [Ext] handlerWriteOption #
    - [Ext] handlerIssueAndWriteOption #
    - [Ext] handlerBuybackOption #
    - [Ext] resetEphemeralValues #
    - [Ext] pauseTradingAndRequest #
    - [Ext] executeEpochCalculation #
       - modifiers: whenNotPaused
    - [Ext] deposit #
       - modifiers: whenNotPaused,nonReentrant
    - [Ext] redeem #
       - modifiers: nonReentrant
```

```
   - [Ext] initiateWithdraw #
      - modifiers: whenNotPaused,nonReentrant
   - [Ext] completeWithdraw #
      - modifiers: whenNotPaused,nonReentrant
   - [Int] _getNormalizedBalance
   - [Pub] getBalance
   - [Pub] getExternalDelta
   - [Pub] getPortfolioDelta
   - [Ext] quotePriceWithUtilizationGreeks
   - [Int] addUtilizationPremium
   - [Int] applyDeltaPremium
   - [Pub] getImpliedVolatility
   - [Ext] getAssets
   - [Ext] getNAV
   - [Int] _redeem #
   - [Int] _getNAV
   - [Int] _getAssets
   - [Int] _getLiabilities
   - [Pub] checkBuffer
   - [Int] _issue #
   - [Int] _writeOption #
   - [Int] _buybackOption #
   - [Int] _adjustVariables #
   - [Int] _getVolatilityFeed
   - [Int] _getPortfolioValuesFeed
   - [Int] _getAccounting
   - [Int] _getOptionRegistry
   - [Int] _getUnderlyingPrice
   - [Int] _isTradingNotPaused
   - [Int] _isHandler
   - [Int] _isKeeper

+  Accounting (IAccounting)
   - [Pub] <Constructor> #
   - [Int] calculateTokenPrice
   - [Ext] deposit
   - [Ext] redeem
   - [Ext] initiateWithdraw
   - [Ext] completeWithdraw
   - [Ext] executeEpochCalculation
   - [Pub] sharesForAmount
   - [Pub] amountForShares

+  AlphaOptionHandler (AccessControl, ReentrancyGuard)
   - [Pub] <Constructor> #
      - modifiers: AccessControl
   - [Ext] setCustomOrderBounds #
   - [Pub] createOrder #
   - [Ext] createStrangle #
   - [Pub] executeOrder #
      - modifiers: nonReentrant
   - [Pub] executeBuyBackOrder #
      - modifiers: nonReentrant
   - [Ext] executeStrangle #
   - [Int] getOptionRegistry
   - [Int] getPortfolioValuesFeed
   - [Int] _getUnderlyingPrice

+  PriceFeed (AccessControl)
   - [Pub] <Constructor> #
      - modifiers: AccessControl
   - [Pub] addPriceFeed #
   - [Ext] getRate
   - [Ext] getNormalizedRate

+  OptionRegistry (AccessControl)
   - [Pub] <Constructor> #
      - modifiers: AccessControl
   - [Ext] setLiquidityPool #
   - [Ext] setKeeper #
   - [Ext] setHealthThresholds #
   - [Ext] issue #
   - [Ext] open #
   - [Ext] close #
   - [Ext] settle #
   - [Ext] adjustCollateral #
   - [Ext] adjustCollateralCaller #
   - [Ext] wCollatLiquidatedVault #
   - [Ext] registerLiquidatedVault #
   - [Ext] redeem #
   - [Ext] getCollateral
   - [Ext] getOtoken
   - [Pub] checkVaultHealth
   - [Ext] getSeriesAddress
   - [Ext] getSeries
   - [Ext] getSeriesInfo
   - [Pub] getIssuanceHash
   - [Int] getIssuanceHash
   - [Pub] formatStrikePrice
   - [Int] _isLiquidityPool
   - [Int] _isKeeper

+  OptionHandler (Pausable, AccessControl, ReentrancyGuard)
   - [Pub] <Constructor> #
      - modifiers: AccessControl
   - [Ext] setCustomOrderBounds #
   - [Ext] pause #
   - [Ext] unpause #
   - [Ext] addOrRemoveBuybackAddress #
   - [Ext] setMinDeltaForRequest #
   - [Pub] createOrder #
   - [Ext] createStrangle #
   - [Pub] executeOrder #
      - modifiers: nonReentrant
   - [Ext] executeStrangle #
   - [Ext] issueAndWriteOption #
      - modifiers: whenNotPaused,nonReentrant
   - [Ext] issue #
      - modifiers: whenNotPaused,nonReentrant
   - [Ext] writeOption #
      - modifiers: whenNotPaused,nonReentrant
   - [Ext] buybackOption #
      - modifiers: nonReentrant,whenNotPaused
   - [Int] getOptionRegistry
   - [Int] getPortfolioValuesFeed
   - [Int] _getUnderlyingPrice

+  UniswapV3HedgingReactor (IHedgingReactor, AccessControl)
   - [Pub] <Constructor> #
      - modifiers: AccessControl
   - [Ext] changePoolFee #
   - [Ext] setMinAmount #
   - [Ext] setSlippage #
   - [Ext] hedgeDelta #
   - [Ext] withdraw #
   - [Ext] update
   - [Ext] getDelta
   - [Ext] getPoolDenominatedValue
   - [Int] _swapExactOutputSingle #
   - [Int] _swapExactInputSingle #
   - [Int] getUnderlyingPrice

+  PerpHedgingReactor (IHedgingReactor, AccessControl)
   - [Pub] <Constructor> #
      - modifiers: AccessControl
   - [Ext] setHealthFactor #
   - [Ext] setKeeper #
   - [Ext] setSyncOnChange #
   - [Ext] initialiseReactor #
   - [Ext] hedgeDelta #
   - [Ext] withdraw #
   - [Ext] syncAndUpdate #
   - [Pub] sync #
   - [Pub] update #
   - [Ext] getDelta
   - [Ext] getPoolDenominatedValue
   - [Ext] checkVaultHealth
   - [Int] _changePosition #
   - [Int] _isKeeper

+  AlphaPortfolioValuesFeed (AccessControl, IPortfolioValuesFeed)
   - [Pub] <Constructor> #
      - modifiers: AccessControl
   - [Ext] setLiquidityPool #
   - [Ext] setProtocol #
   - [Ext] setRFR #
   - [Ext] setKeeper #
   - [Ext] setHandler #
   - [Ext] fulfill #
   - [Ext] updateStores #
   - [Ext] syncLooper #
   - [Ext] cleanLooperManually #
   - [Int] _cleanLooper #
   - [Ext] accountLiquidatedSeries #
   - [Ext] migrate #
   - [Ext] requestPortfolioData #
   - [Ext] getPortfolioValues
   - [Int] _isKeeper
   - [Int] _isHandler
   - [Ext] isAddressInSet
   - [Ext] addressAtIndexInSet
   - [Ext] addressSetLength
   - [Ext] getAddressSet
   - [Int] _getVolatilityFeed
   - [Int] _getOptionRegistry
   - [Int] _getUnderlyingPrice

+  PortfolioValuesFeed (AccessControl, ChainlinkClient)
   - [Pub] <Constructor> #
      - modifiers: AccessControl
   - [Ext] setOracle #
   - [Ext] setLiquidityPool #
   - [Ext] setAddressStringMapping #
   - [Ext] setLink #
   - [Ext] setKeeper #
   - [Ext] fulfill #
      - modifiers: recordChainlinkFulfillment
   - [Ext] withdrawLink #
   - [Ext] requestPortfolioData #
   - [Ext] getPortfolioValues
   - [Int] _isKeeper


($) = payable function
# = non-constant function
```

Test

```
❯ npm test

> delta-hedging@1.0.0 test
> hardhat test
```

```
(node:26732) Warning: Accessing non-existent property 'INVALID_ALT_NUMBER' of module exports inside circul
(Use `node --trace-warnings ...` to show where the warning was created)
(node:26732) Warning: Accessing non-existent property 'INVALID_ALT_NUMBER' of module exports inside circul
No need to generate any newer typings.
 .-------------------------------------|-------------|---------------.
 |  Contract Name                      ·  Size (KB)  ·  Change (KB)  |
 ......................................|.............|................
 |  ReentrancyGuard                    ·      0.063  ·               |
 ......................................|.............|................
 |  OtokenSpawner                      ·      0.063  ·               |
 ......................................|.............|................
 |  Spawn                              ·      0.063  ·               |
 ......................................|.............|................
 |  ReentrancyGuardUpgradeSafe         ·      0.063  ·               |
 ......................................|.............|................
 |  Initializable                      ·      0.063  ·               |
 ......................................|.............|................
 |  console                            ·      0.086  ·               |
 ......................................|.............|................
 |  GammaTypes                         ·      0.086  ·               |
 ......................................|.............|................
 |  SafeTransferLib                    ·      0.086  ·               |
 ......................................|.............|................
 |  Types                              ·      0.086  ·               |
 ......................................|.............|................
 |  EnumerableSet                      ·      0.086  ·               |
 ......................................|.............|................
 |  PRBMath                            ·      0.086  ·               |
 ......................................|.............|................
 |  PRBMathUD60x18                     ·      0.086  ·               |
 ......................................|.............|................
 |  PRBMathSD59x18                     ·      0.086  ·               |
 ......................................|.............|................
 |  AddressUpgradeable                 ·      0.086  ·               |
 ......................................|.............|................
 |  Address                            ·      0.086  ·               |
 ......................................|.............|................
 |  StorageSlot                        ·      0.086  ·               |
 ......................................|.............|................
 |  Strings                            ·      0.086  ·               |
 ......................................|.............|................
 |  Create2                            ·      0.086  ·               |
 ......................................|.............|................
 |  Chainlink                          ·      0.086  ·               |
 ......................................|.............|................
 |  EnumerableSet                      ·      0.086  ·               |
 ......................................|.............|................
 |  ClearingHouseExtsload              ·      0.086  ·               |
 ......................................|.............|................
 |  Bisection                          ·      0.086  ·               |
 ······································|·············|················
 |  Math                               ·      0.086  ·               |
 ······································|·············|················
 |  BatchedLoop                        ·      0.086  ·               |
 ······································|·············|················
 |  FundingPayment                     ·      0.086  ·               |
 ······································|·············|················
 |  CollateralDeposit                  ·      0.086  ·               |
 ······································|·············|················
 |  SignedFullMath                     ·      0.086  ·               |
 ······································|·············|················
 |  GoodAddressDeployer                ·      0.086  ·               |
 ······································|·············|················
 |  TickBitmapExtended                 ·      0.086  ·               |
 ······································|·············|················
 |  SignedMath                         ·      0.086  ·               |
 ······································|·············|················
 |  AddressHelper                      ·      0.086  ·               |
 ······································|·············|················
 |  SafeCast                           ·      0.086  ·               |
 ······································|·············|················
 |  SimulateSwap                       ·      0.086  ·               |
 ······································|·············|················
 |  SwapMath                           ·      0.086  ·               |
 ······································|·············|················
 |  LiquidityPositionSet               ·      0.086  ·               |
 ······································|·············|················
 |  Uint32L8ArrayLib                   ·      0.086  ·               |
 ······································|·············|················
 |  LiquidityPosition                  ·      0.086  ·               |
 ······································|·············|················
 |  Block                              ·      0.086  ·               |
 ······································|·············|················
 |  TickExtended                       ·      0.086  ·               |
 ······································|·············|················
 |  PriceMath                          ·      0.086  ·               |
 ······································|·············|················
 |  Uint48Lib                          ·      0.086  ·               |
 ······································|·············|················
 |  VTokenPosition                     ·      0.086  ·               |
 ······································|·············|················
 |  Uint48L5ArrayLib                   ·      0.086  ·               |
 ······································|·············|················
 |  UniswapV3PoolHelper                ·      0.086  ·               |
 ······································|·············|················
 |  WordHelper                         ·      0.086  ·               |
 ······································|·············|················
 |  VTokenPositionSet                  ·      0.086  ·               |
 ······································|·············|················
 |  FixedPoint128                      ·      0.086  ·               |
 ······································|·············|················
```

```
|  Protocol                           ·      0.086  ·               |
······································|·············|················
|  SqrtPriceMath                      ·      0.086  ·               |
······································|·············|················
|  FixedPoint96                       ·      0.086  ·               |
······································|·············|················
|  FullMath                           ·      0.086  ·               |
······································|·············|················
|  TickMath                           ·      0.086  ·               |
······································|·············|················
|  SwapMath                           ·      0.086  ·               |
······································|·············|················
|  BitMath                            ·      0.086  ·               |
······································|·············|················
|  UnsafeMath                         ·      0.086  ·               |
······································|·············|················
|  SafeCast                           ·      0.086  ·               |
······································|·············|················
|  FixedPointInt256                   ·      0.086  ·               |
······································|·············|················
|  SignedConverter                    ·      0.086  ·               |
······································|·············|················
|  Actions                            ·      0.086  ·               |
······································|·············|················
|  BokkyPooBahsDateTimeLibrary        ·      0.086  ·               |
······································|·············|················
|  BufferChainlink                    ·      0.086  ·               |
······································|·············|················
|  CBORChainlink                      ·      0.086  ·               |
······································|·············|················
|  SafeERC20                          ·      0.086  ·               |
······································|·············|················
|  Create2                            ·      0.086  ·               |
······································|·············|················
|  SafeMath                           ·      0.086  ·               |
······································|·············|················
|  Strings                            ·      0.086  ·               |
······································|·············|················
|  SafeERC20                          ·      0.086  ·               |
······································|·············|················
|  Address                            ·      0.086  ·               |
······································|·············|················
|  SignedSafeMath                     ·      0.086  ·               |
······································|·············|················
|  SafeMathUpgradeable                ·      0.086  ·               |
······································|·············|················
|  ECDSAUpgradeable                   ·      0.086  ·               |
······································|·············|················
|  CountersUpgradeable                ·      0.086  ·               |
······································|·············|················
|  ForceSend                          ·      0.162  ·               |
······································|·············|················
|  SettlementTokenOracle              ·      0.180  ·               |
······································|·············|················
|  Migrations                         ·      0.253  ·               |
······································|·············|················
|  UpgradeabilityProxy                ·      0.262  ·               |
······································|·············|················
|  SignedConverterTester              ·      0.380  ·               |
······································|·············|················
|  CallTester                         ·      0.469  ·               |
······································|·············|················
|  MockPricer                         ·      0.529  ·               |
······································|·············|················
|  BlackScholesTest                   ·      0.546  ·               |
······································|·············|················
|  UpgradeableContractV1              ·      0.552  ·               |
······································|·············|················
|  UpgradeableContractV2              ·      0.576  ·               |
······································|·············|················
|  MockChainlinkAggregator            ·      0.611  ·               |
······································|·············|················
|  ERC1967Proxy                       ·      0.680  ·               |
······································|·············|················
|  PermitCallee                       ·      0.690  ·               |
······································|·············|················
|  OwnableUpgradeSafe                 ·      0.712  ·               |
······································|·············|················
|  OptionsCompute                     ·      0.799  ·               |
······································|·············|················
|  MockAddressBook                    ·      0.988  ·               |
······································|·············|················
|  Protocol                           ·      1.004  ·               |
······································|·············|················
|  MockController                     ·      1.040  ·               |
······································|·············|················
|  Mock0xERC20Proxy                   ·      1.052  ·               |
······································|·············|················
|  FlashUnwrap                        ·      1.095  ·               |
······································|·············|················
|  Volatility                         ·      1.145  ·               |
······································|·············|················
|  UniswapV3HedgingTest               ·      1.153  ·               |
······································|·············|················
|  OwnedUpgradeabilityProxy           ·      1.173  ·               |
······································|·············|················
|  OracleMock                         ·      1.217  ·               |
······································|·············|················
|  MockWhitelistModule                ·      1.297  ·               |
······································|·············|················
|  CalleeAllowanceTester              ·      1.298  ·               |
······································|·············|················
```

```
|  PerpHedgingTest                    ·      1.344  ·               |
······································|·············|················
|  WstethPricer                       ·      1.480  ·               |
······································|·············|················
|  YearnPricer                        ·      1.710  ·               |
······································|·············|················
|  Authority                          ·      1.718  ·               |
······································|·············|················
|  FixedPointInt256Tester             ·      1.749  ·               |
······································|·············|················
|  CompoundPricer                     ·      1.806  ·               |
······································|·············|················
|  ProxyAdmin                         ·      1.827  ·               |
······································|·············|················
|  WETH                               ·      1.900  ·               |
······································|·············|················
|  MarginVaultTester                  ·      1.983  ·               |
······································|·············|················
|  ChainLinkPricer                    ·      1.999  ·               |
······································|·············|················
|  WETH9                              ·      2.026  ·               |
······································|·············|················
|  TransparentUpgradeableProxy        ·      2.064  ·               |
······································|·············|················
|  PriceFeed                          ·      2.089  ·               |
······································|·············|················
|  Mock0xExchange                     ·      2.112  ·               |
······································|·············|················
|  MockDumbERC20                      ·      2.154  ·               |
······································|·············|················
|  LiquidityPoolAdjustCollateralTest  ·      2.204  ·               |
······································|·············|················
|  ERC20                              ·      2.243  ·               |
······································|·············|················
|  ERC20Upgradeable                   ·      2.253  ·               |
······································|·············|················
|  ERC20Upgradeable                   ·      2.578  ·               |
······································|·············|················
|  MockOracle                         ·      2.579  ·               |
······································|·············|················
|  MarginVault                        ·      2.701  ·               |
······································|·············|················
|  MockERC20                          ·      2.903  ·               |
······································|·············|················
|  MockCToken                         ·      3.005  ·               |
······································|·············|················
|  MockWSTETHToken                    ·      3.005  ·               |
······································|·············|················
|  MockYToken                         ·      3.005  ·               |
······································|·············|················
|  MockPortfolioValuesFeed            ·      3.219  ·               |
······································|·············|················
|  VolatilityFeed                     ·      3.250  ·               |
······································|·············|················
|  Whitelist                          ·      3.579  ·               |
······································|·············|················
|  VToken                             ·      3.781  ·               |
······································|·············|················
|  NormalDist                         ·      3.794  ·               |
······································|·············|················
|  VQuote                             ·      3.802  ·               |
······································|·············|················
|  PayableProxyController             ·      4.016  ·               |
······································|·············|················
|  MockCUSDC                          ·      4.186  ·               |
······································|·············|················
|  MockPermitERC20                    ·      4.327  ·               |
······································|·············|················
|  NewWhitelist                       ·      4.336  ·               |
······································|·············|················
|  OtokenFactory                      ·      4.611  ·               |
······································|·············|················
|  AddressBook                        ·      4.992  ·               |
······································|·············|················
|  InsuranceFund                      ·      5.410  ·               |
······································|·············|················
|  ActionTester                       ·      5.648  ·               |
······································|·············|················
|  MarginPool                         ·      5.689  ·               |
······································|·············|················
|  Accounting                         ·      5.847  ·               |
······································|·············|················
|  Oracle                             ·      5.882  ·               |
······································|·············|················
|  UniswapV3HedgingReactor            ·      6.263  ·               |
······································|·············|················
|  RealTokenMock                      ·      6.539  ·               |
······································|·············|················
|  ERC20PresetMinterPauser            ·      6.539  ·               |
······································|·············|················
|  BlackScholes                       ·      6.739  ·               |
······································|·············|················
|  PortfolioValuesFeed                ·      7.024  ·               |
······································|·············|················
|  OtokenImplV1                       ·      7.055  ·               |
······································|·············|················
|  MockOtoken                         ·      7.212  ·               |
······································|·············|················
|  OpynInteractions                   ·      7.225  ·               |
······································|·············|················
|  AlphaOptionHandler                 ·      8.854  ·               |
······································|·············|················
```

```
|  AlphaPortfolioValuesFeed           ·      8.916  ·               |
······································|·············|················
|  ClearingHouseLens                  ·      9.564  ·               |
······································|·············|················
|  PerpHedgingReactor                 ·     11.917  ·               |
······································|·············|················
|  Otoken                             ·     12.169  ·               |
······································|·············|················
|  OptionHandler                      ·     13.310  ·               |
······································|·············|················
|  RageTradeFactory                   ·     14.406  ·               |
······································|·············|················
|  OptionRegistry                     ·     15.284  ·               |
······································|·············|················
|  MarginCalculator                   ·     17.179  ·               |
······································|·············|················
|  CalculatorTester                   ·     17.673  ·               |
······································|·············|················
|  NewMarginCalculator                ·     18.137  ·               |
······································|·············|················
|  VPoolWrapper                       ·     19.965  ·               |
······································|·············|················
|  ClearingHouse                      ·     23.020  ·               |
······································|·············|················
|  Account                            ·     23.184  ·               |
······································|·············|················
|  Controller                         ·     23.951  ·               |
······································|·············|················
|  LiquidityPool                      ·     24.117  ·               |
······································|·············|················
|  NewController                      ·     24.436  ·               |
·-------------------------------------|-------------|---------------


 Pricing options
   ✓ Should deploy Black Scholes library
   ✓ correctly prices in the money call with a one year time to expiration
   ✓ correctly  prices out of the money call with one year time
   ✓ correctly prices out of the money call with one year time high volatility
   ✓ correctly prices in the money call with one month expiration high volatility
   ✓ correctly prices in the money put with one year time
   ✓ correctly prices in the money put with one year time high volatility
   ✓ correctly prices in the money put with one month time high volatility
   ✓ correctly prices in the money put with one month time high volatility
   ✓ correctly prices at the money put with one month time high volatility
   ✓ correctly prices near the money put with one month time high volatility
   ✓ correctly prices out of the money put with one month time high volatility
   ✓ correctly prices out of the money put with one month time
   ✓ correctly computes delta of out of the money call with one month time
   ✓ correctly computes delta of out of the money put with one month time
   ✓ Estimated portfolio deltas should deviate by more than 10% compared with cached values at scale
```

```
  Authority tests
    Authority push effective immediately
      ✓ SUCCEEDS: set governor
      ✓ SUCCEEDS: set manager
      ✓ SUCCEEDS: set guardian
      ✓ SUCCEEDS: revoke guardian
      ✓ FAILS: revoke guardian when not auth
      ✓ FAILS: set governor when not auth
      ✓ FAILS: set manager when not auth
      ✓ FAILS: set guardian when not auth
    Authority push and pull
      ✓ SUCCEEDS: push governor
      ✓ FAILS: rando tries to pull governor rank
      ✓ SUCCEEDS: pull governor rank
      ✓ SUCCEEDS: push manager
      ✓ FAILS: rando tries to pull manager rank
      ✓ SUCCEEDS: pull manager rank

  Dyn Quote Tests
    ✓ Deposit to the liquidityPool
    Quote
      ✓ gets price
      ✓ Returns a quote for a ETH/USD put with utilization
      ✓ Returns a quote for ETH/USD call with utilization
      ✓ Returns a quote for a ETH/USD put to buy
      ✓ Returns a quote for ETH/USD call to buy

  Hegic Attack
    ✓ Adds liquidity to the liquidityPool
    ✓ Attacker adds liquidity
    ✓ pauses trading and executes epoch
    ✓ LP Writes a WETH/USD put collateralized by USD for premium to the attacker
    ✓ attacker initiates withdraw liquidity
    ✓ pauses trading and executes epoch
    ✓ attacker withdraws liquidity

  RR oracle between update attack vector
    Sc 1. Single large option purchase and update checks
      ✓ Sc1. Adds liquidity to the liquidityPool
      ✓ Sc1. Another adds liquidity to the liquidityPool
      ✓ Sc1. pauses trading and executes epoch
{ collateralAllocatedBefore: BigNumber { value: "0" } }
{ quote: '356055.638504510656052' }
Pool should now have delta value of: 222850242957618327600
Pool should now have an options portfolio value of (or liabilities): 356055.60262349993
NAV after issuance should be: 2000000000000000000000000
NAV after issuance is: 2000000000000000000000000
{ collateralAllocatedAfter: BigNumber { value: "1086994811040" } }
      ✓ Sc1. LP Writes a WETH/USD put collateralized by USD for premium to the attacker
{
  utilizationBefore: 0,
  utilizationAfter: 0.5434974055,
  utilizationPrice: 323686.8510865387
}
{
  utilizationBefore: 0,
  utilizationAfter: 0.5434974055,
  utilizationPrice: 323686.8510865387
}
{
  portfolioDelta: 222.8502733230175,
  portfolioGamma: -0.12854324091650596,
  portfolioTheta: 2580.4147415762804,
  portfolioVega: -1373.1578070350035,
  callsPutsValue: 356055.53619519254,
  bsCallsPutsValue: 323686.8510865387
}
{
  beforeNAV: BigNumber { value: "2000000000000000000000000" },
  afterNAV: BigNumber { value: "2000000036124807460000000" }
}
{ collateralAllocated: BigNumber { value: "1086994811040" } }
      ✓ Sc1. should update NAV after fulfill
      ✓ Sc1. initiates withdraw liquidity
{
  utilizationBefore: 0,
  utilizationAfter: 0.5434974055,
  utilizationPrice: 323686.76083484804
}
{
  utilizationBefore: 0,
  utilizationAfter: 0.5434974055,
  utilizationPrice: 323686.76083484804
}
{
  portfolioDelta: 222.85031900070163,
  portfolioGamma: -0.1285432971268954,
  portfolioTheta: 2580.4158811195616,
  portfolioVega: -1373.1570975062339,
  callsPutsValue: 356055.4369183328,
  bsCallsPutsValue: 323686.76083484804
}
      ✓ Sc1. pauses trading and executes epoch
liabilities are now 0 because the pool isnt updated
USDC withdrawn: 1000000067700
NAV after withdraw should be: 1000000000000000000000000
NAV after withdraw is: 1000000067701667200000000
{
  utilizationBefore: 0,
  utilizationAfter: 1.086994884915652,
  utilizationPrice: 533019.687967529
}
```

```
{
  utilizationBefore: 0,
  utilizationAfter: 1.086994884915652,
  utilizationPrice: 533019.687967529
}
{
  portfolioDelta: 222.85036467843233,
  portfolioGamma: -0.1285433533373513,
  portfolioTheta: 2580.4170206641825,
  portfolioVega: -1373.1563879770351,
  callsPutsValue: 586321.6567642819,
  bsCallsPutsValue: 323686.67058311775
}
NAV after update should be: 1000000000000000000000000
NAV after update is: 769733847855718100000000
      ✓ Sc1. attacker withdraws liquidity before delta and portfolio values update
    Sc 2. Two Seperate Single large option purchase and update checks
      ✓ Sc2. Adds liquidity to the liquidityPool
      ✓ Sc2. Another adds liquidity to the liquidityPool
      ✓ Sc2. pauses trading and executes epoch
{ collateralAllocatedBefore: BigNumber { value: "0" } }
{ quote: '356055.638504510656052' }
Pool should now have delta value of: 222850242957618327600
Pool should now have an options portfolio value of (or liabilities): 356055.60262349993
NAV after issuance should be: 2000000000000000000000000
NAV after issuance is: 2000000000000000000000000
{ collateralAllocatedAfter: BigNumber { value: "1086994811040" } }
      ✓ Sc2. LP Writes a WETH/USD put collateralized by USD for premium to the attacker
{ collateralAllocatedBefore: BigNumber { value: "1086994811040" } }
{ quote: '67253.98515438584210283' }
Pool should now have delta value of: 297851648003624958200
Pool should now have an options portfolio value of (or liabilities): 423309.56377159094
NAV after issuance should be: 2000000000000000000000000
NAV after issuance is: 2000000000000000000000000
{ collateralAllocatedAfter: BigNumber { value: "1339664439167" } }
      ✓ Sc2. LP Writes a WETH/USD call collateralized by USD for premium to the attacker
{
  utilizationBefore: 0,
  utilizationAfter: 0.5990323976921558,
  utilizationPrice: 323686.76083484804
}
{
  utilizationBefore: 0.46136212741250227,
  utilizationAfter: 0.5686047710083469,
  utilizationPrice: 74726.61057647658
}
{
  utilizationBefore: 0,
  utilizationAfter: 0.5990323976921558,
  utilizationPrice: 323686.76083484804
}
```

```
{
  utilizationBefore: 0.46136212741250227,
  utilizationAfter: 0.5686047710083469,
  utilizationPrice: 74726.61057647658
}
{
  portfolioDelta: 147.8489118554822,
  portfolioGamma: -0.1538577842485459,
  portfolioTheta: 3137.450602070023,
  portfolioVega: -1649.4325030842474,
  callsPutsValue: 406959.5777684639,
  bsCallsPutsValue: 398413.3714113246
}
{
  beforeNAV: BigNumber { value: "200000000000000000000000" },
  afterNAV: BigNumber { value: "201634996803453610000000" }
}
{ collateralAllocated: BigNumber { value: "1339664439167" } }
      ✓ Sc2. should update NAV after fulfill
      ✓ Sc2. initiates withdraw liquidity
{
  utilizationBefore: 0,
  utilizationAfter: 0.5990323976921558,
  utilizationPrice: 323686.67058311775
}
{
  utilizationBefore: 0.46136212741250227,
  utilizationAfter: 0.5686047710083469,
  utilizationPrice: 74726.59112496504
}
{
  utilizationBefore: 0,
  utilizationAfter: 0.5990323976921558,
  utilizationPrice: 323686.67058311775
}
{
  utilizationBefore: 0.46136212741250227,
  utilizationAfter: 0.5686047710083469,
  utilizationPrice: 74726.59112496504
}
{
  portfolioDelta: 147.8489544204138,
  portfolioGamma: -0.15385785089292284,
  portfolioTheta: 3137.4519656818993,
  portfolioVega: -1649.4316440253403,
  callsPutsValue: 406959.47421928355,
  bsCallsPutsValue: 398413.2617080828
}
NAV after withdraw is: 201635007158371645000000
      ✓ Sc2. pauses trading and executes epoch
    Sc 3. Single large option purchase and update and another option purchase checks
```

```
      ✓ Sc3. Adds liquidity to the liquidityPool
      ✓ Sc3. Another adds liquidity to the liquidityPool
      ✓ Sc3. pauses trading and executes epoch
{ collateralAllocatedBefore: BigNumber { value: "0" } }
{ quote: '356055.638504510656052' }
Pool should now have delta value of: 222850242957618327600
Pool should now have an options portfolio value of (or liabilities): 356055.60262349993
NAV after issuance should be: 2000000000000000000000000
NAV after issuance is: 2000000000000000000000000
{ collateralAllocatedAfter: BigNumber { value: "1086994811040" } }
      ✓ Sc3. LP Writes a WETH/USD put collateralized by USD for premium to the attacker
{
  utilizationBefore: 0,
  utilizationAfter: 0.5434974055,
  utilizationPrice: 323686.8510865387
}
{
  utilizationBefore: 0,
  utilizationAfter: 0.5434974055,
  utilizationPrice: 323686.8510865387
}
{
  portfolioDelta: 222.8502733230175,
  portfolioGamma: -0.12854324091650596,
  portfolioTheta: 2580.4147415762804,
  portfolioVega: -1373.1578070350035,
  callsPutsValue: 356055.53619519254,
  bsCallsPutsValue: 323686.8510865387
}
{
  beforeNAV: BigNumber { value: "2000000000000000000000000" },
  afterNAV: BigNumber { value: "2000000036124807460000000" }
}
{ collateralAllocated: BigNumber { value: "1086994811040" } }
      ✓ Sc3. should update NAV after fulfill
{ collateralAllocatedBefore: BigNumber { value: "1086994811040" } }
{ quote: '67253.97931894967072049' }
Pool should now have delta value of: 297851649041222099700
Pool should now have an options portfolio value of (or liabilities): 423309.5579361398
NAV after issuance should be: 2000000000000000000000000
NAV after issuance is: 2000000036124807460000000
{ collateralAllocatedAfter: BigNumber { value: "1339664439167" } }
      ✓ Sc3. LP Writes a WETH/USD call collateralized by USD for premium to the attacker
      ✓ Sc3. initiates withdraw liquidity
{
  utilizationBefore: 0,
  utilizationAfter: 0.5990323996728818,
  utilizationPrice: 323686.67058311775
}
{
  utilizationBefore: 0.46136212741250227,
  utilizationAfter: 0.5686047710083469,
  utilizationPrice: 74726.59112496504
}
{
  utilizationBefore: 0,
  utilizationAfter: 0.5990323996728818,
  utilizationPrice: 323686.67058311775
}
{
  utilizationBefore: 0.46136212741250227,
  utilizationAfter: 0.5686047710083469,
  utilizationPrice: 74726.59112496504
}
{
  portfolioDelta: 147.8489544204138,
  portfolioGamma: -0.15385785089292284,
  portfolioTheta: 3137.4519656818993,
  portfolioVega: -1649.4316440253403,
  callsPutsValue: 406959.47421928355,
  bsCallsPutsValue: 398413.2617080828
}
NAV after withdraw is: 201635006574871645000000
      ✓ Sc2. pauses trading and executes epoch

  Liquidity Pools
    ✓ Succeeds: sets utilization skew params correctly
    ✓ Succeeds: User 1: Deposit to the liquidityPool
    ✓ Succeeds: pauses trading
    ✓ Succeeds: execute epoch
    ✓ deploys the hedging reactor
    ✓ sets reactor address on LP contract
    ✓ Returns a quote for a ETH/USD put with utilization
    ✓ Returns a quote for a ETH/USD put to buy
    ✓ Reverts: Push to price deviation threshold to cause quote to fail
    ✓ Reverts: Push to price deviation threshold to cause quote to fail other way
    ✓ Reverts: Push to time deviation threshold to cause quote to fail
    ✓ reverts when attempting to write ETH/USD puts with expiry outside of limit
    ✓ reverts when attempting to write a ETH/USD put with strike outside of limit
    ✓ reverts when attempting to write ETH/USD call with expiry outside of limit
    ✓ reverts when attempting to write a ETH/USD call with strike outside of limit
    ✓ can compute portfolio delta
    ✓ LP Writes a ETH/USD put for premium
    ✓ can issue a put series
    ✓ can issue a call series
    ✓ can compute portfolio delta
    ✓ writes more options for an existing series
    ✓ pauses and unpauses handler contract
    ✓ LP writes another ETH/USD put that expires later
    ✓ adds address to the buyback whitelist
    ✓ LP can buy back option to reduce open interest
    ✓ fails if buyback token address is invalid
    ✓ buys back an option from a non-whitelisted address if it moves delta closer to zero
    ✓ can compute portfolio delta
    ✓ reverts if option collateral exceeds buffer limit
    ✓ reverts when non-admin calls rebalance function
    ✓ reverts when rebalance delta too small
    ✓ returns zero when hedging positive delta when reactor has no funds
    ✓ Returns a quote for ETH/USD call with utilization
    ✓ Creates a buy order
    ✓ creates a custom strangle order
    ✓ Cant make a buy order if not admin
    ✓ Create buy order reverts if price is zero
    ✓ Create buy order reverts if order expiry too long
    ✓ cant exercise order if not buyer
    ✓ Executes a buy order
    ✓ executes a strangle
    ✓ does not buy back an option from a non-whitelisted address if it moves delta away to zero
    ✓ Cannot complete buy order after expiry
    ✓ fails to execute invalid custom orders
    ✓ Can compute IV from volatility skew coefs
    ✓ Succeeds: User 2: Deposit to the liquidityPool
    ✓ Succeeds: pauses trading
    ✓ Succeeds: execute epoch
    ✓ Succeed: User 1: redeems all shares
    ✓ Succeed: User 1: Initiates Withdraw for half owned balance
    ✓ pauses and unpauses LP contract
    ✓ settles an expired ITM vault
    ✓ settles an expired OTM vault
    ✓ Reverts: tries to sell an expired option back to the pool
    ✓ Reverts: tries to write an option that doesnt exist in the handler
    ✓ updates option params with setter
    ✓ adds and deletes a hedging reactor address
    ✓ sets new custom order bounds
    ✓ updates collateralCap variable
    ✓ updates maxDiscount variable
    ✓ updates bufferPercentage variable
    ✓ updates riskFreeRate variable
    ✓ sets new utilization skew params
    ✓ pauses trading
    ✓ handler-only functions in Liquidity pool revert if not called by handler
    ✓ returns a volatility skew
    ✓ protocol changes feeds

  Liquidity Pool with alpha tests
    Deposit funds into the liquidityPool
      ✓ SUCCEEDS: User 1: Deposit to the liquidityPool
      ✓ SUCCEEDS: pauses trading
      ✓ Succeeds: execute epoch
    Create and execute a single buy order
      ✓ SUCCEEDS: Creates a buy order
      ✓ REVERTS: Cant make a buy order if not admin
      ✓ REVERTS: Cant create buy order if price is zero
      ✓ REVERTS: Cant create buy order if order expiry too long
      ✓ REVERTS: cant exercise order if not buyer
      ✓ REVERTS: Cant execute sell order to buyback order
      ✓ SUCCEEDS: Executes a buy order
    Create and execute a strangle
      ✓ SUCCEEDS: creates a custom strangle order
      ✓ SETUP: fulfill
      ✓ SUCCEEDS: executes a strangle
    Create and execute a single buyback order
      ✓ SETUP: Creates a buy order
      ✓ SETUP: Executes a buy order
      ✓ SUCCEEDS: Creates a buyback order
      ✓ REVERTS: Cant make a buyback order if not admin
      ✓ REVERTS: Cant create buyback order if price is zero
      ✓ REVERTS: Cant create buyback order if order expiry too long
      ✓ REVERTS: cant exercise order if not buyer
      ✓ REVERTS: Cant execute buyback order to sell order
      ✓ SUCCEEDS: Executes a buyback order
      ✓ SUCCEEDS: Creates a buyback order on the same option
      ✓ REVERTS: Doesnt Execute a buyback order for option with no position
    Create a buy order and fail to meet order in time
      ✓ SUCCEEDS: Creates a buy order
      ✓ REVERTS: Cant execute after order expires
    Create a buy order and spot moves past deviation threshold
      ✓ SUCCEEDS: Creates a buy order
      ✓ REVERTS: Cant execute after spot moves too much up
      ✓ REVERTS: Cant execute after spot moves too much down
    Liquidate a position and update stores, make sure stores update properly
      ✓ SETUP: partially liquidates a vault
      ✓ SUCCEEDS: sets stores to correct amount of liquidated vault
      ✓ REVERTS: cant account series that isnt stored
    Deposit funds into the liquidityPool and withdraw
      ✓ SUCCEEDS: User 2: Deposit to the liquidityPool
      ✓ SUCCEEDS: pauses trading
      ✓ Succeeds: execute epoch
      ✓ SUCCEEDS: User 1: redeems all shares
      ✓ SUCCEEDS: User 1: Initiates Withdraw for half owned balance


  Liquidity Pools Deposit Withdraw
    Deposit funds into the liquidityPool
      ✓ Succeeds: User 1: Deposit to the liquidityPool
      ✓ Succeeds: User 1: Deposit to the liquidityPool again
      ✓ Succeeds: User 2: Deposit to the liquidityPool
      ✓ Reverts: User 1: Tries Zero on all functions
      ✓ Reverts: User 1: Attempts to redeem before epoch initiation
      ✓ Reverts: User 1: Attempts to initiate withdraw before epoch initiation
      ✓ Reverts: User 1: Attempts to complete withdraw before epoch initiation
      ✓ Reverts: execute epoch before pause
      ✓ Succeeds: pauses trading
      ✓ Succeeds: execute epoch
    Create and execute a single buy order
```

```
      ✓ SUCCEEDS: Creates a buy order
      ✓ REVERTS: Cant make a buy order if not admin
      ✓ REVERTS: Cant create buy order if price is zero
      ✓ REVERTS: Cant create buy order if order expiry too long
      ✓ REVERTS: cant exercise order if not buyer
      ✓ SUCCEEDS: Executes a buy order
    has another deposit
      ✓ Succeeds: User 3: Deposit to the liquidityPool
    Users redeem their shares
      ✓ Reverts: User 3: Attempts to redeem before epoch initiation
      ✓ Reverts: User 3: Attempts to initiate withdraw before epoch initiation
      ✓ Reverts: User 3: Attempts to complete withdraw before epoch initiation
      ✓ Succeed: User 1: redeems all shares
      ✓ Revert: User 1: redeems all shares again
      ✓ Succeed: User 2: redeems partial shares
    user initiates withdraw their funds
      ✓ Succeed: User 1: Initiates Withdraw for half owned balance
    Create and execute a strangle
      ✓ SUCCEEDS: creates a custom strangle order
      ✓ SETUP: fulfill
      ✓ SUCCEEDS: executes a strangle
    executes epoch with new position
      ✓ Succeeds: pauses trading
      ✓ Succeeds: execute epoch
    more users deposit/withdraw
      ✓ Succeeds: User 3: Deposit to the liquidityPool
      ✓ Succeeds: User 1: can complete withdrawal
      ✓ Succeed: User 1: Initiates Withdraw for half owned balance
      ✓ Succeed: User 2: Initiates Withdraw for owned balance with same redeemable balance
      ✓ Succeed: User 2: Initiates Withdraw for owned balance again in same epoch (not using redeemable sh
      ✓ Reverts: User 1: cannot complete withdrawal because of epoch not closed
      ✓ Succeeds: pauses trading
      ✓ Succeeds: execute epoch

  Liquidity Pools Deposit Withdraw
    ✓ Succeeds: User 1: Deposit to the liquidityPool
    ✓ Succeeds: User 1: Deposit to the liquidityPool again
    ✓ Succeeds: User 2: Deposit to the liquidityPool
    ✓ Reverts: User 1: Tries Zero on all functions
    ✓ Reverts: User 1: Attempts to redeem before epoch initiation
    ✓ Reverts: User 1: Attempts to initiate withdraw before epoch initiation
    ✓ Reverts: User 1: Attempts to complete withdraw before epoch initiation
    ✓ Reverts: execute epoch before pause
    ✓ Succeeds: pauses trading
    ✓ Succeeds: User 1: issues an option
    ✓ Succeeds: execute epoch
    ✓ Succeeds: User 3: Deposit to the liquidityPool
    ✓ Reverts: User 3: Attempts to redeem before epoch initiation
    ✓ Reverts: User 3: Attempts to initiate withdraw before epoch initiation
    ✓ Reverts: User 3: Attempts to complete withdraw before epoch initiation
    ✓ Succeed: User 1: redeems all shares
    ✓ Revert: User 1: redeems all shares again
    ✓ Succeed: User 2: redeems partial shares
    ✓ Succeed: User 1: Initiates Withdraw for half owned balance
    ✓ Succeeds: User 1: LP Writes a ETH/USD put for premium
    ✓ Succeeds: pauses trading
    ✓ Reverts: User 1: cant write option
    ✓ Reverts: User 1: cant issue and write option
    ✓ Succeeds: execute epoch
    ✓ Succeeds: User 3: Deposit to the liquidityPool
    ✓ Succeeds: User 1: can complete withdrawal
    ✓ Succeed: User 1: Initiates Withdraw for half owned balance
    ✓ Succeed: User 2: Initiates Withdraw for owned balance with same redeemable balance
    ✓ Succeed: User 2: Initiates Withdraw for owned balance again in same epoch (not using redeemable shar
    ✓ Succeeds: User 1: LP Writes a ETH/USD put for premium
    ✓ Reverts: User 1: cannot complete withdrawal because of epoch not closed
    ✓ Succeeds: pauses trading
    ✓ Succeeds: execute epoch with not enough funds to execute withdrawals
    ✓ Reverts: User 1: still cannot complete withdrawal because of withdrawal epoch not closed
    ✓ Succeeds: Reduces collateral cap
    ✓ Reverts: User 1: Deposit to the liquidityPool but hits collat cap
    ✓ Succeeds: Raises collateral cap
    ✓ Succeeds: pauses trading from keeper
    ✓ Succeeds: execute epoch from keeper
    ✓ Reverts: pauses trading from unauthorised
    ✓ Reverts: execute epoch from unauthorised

  Liquidity Pools hedging reactor: perps
    ✓ Deposit to the liquidityPool
    ✓ pauses trading and executes epoch
    ✓ #deploys rage
    ✓ #deploys the hedging reactor
    ✓ #deploy range order
    ✓ can compute portfolio delta
    ✓ LP Writes a ETH/USD put for premium
    ✓ LP writes another ETH/USD put that expires later
    ✓ can compute portfolio delta
    ✓ reverts when non-admin calls rebalance function
    ✓ hedges positive delta in perp hedging reactor
    ✓ Adds additional liquidity from new account
    ✓ pauses trading and executes epoch
    ✓ initiates withdraw liquidity
    ✓ pauses trading and executes epoch
    ✓ LP can redeem shares
    ✓ settles an expired ITM vault
    ✓ settles an expired OTM vault
    ✓ Succeed: Perp hedging reactor unwind

  Liquidity Pools hedging reactor: univ3
    ✓ Deposit to the liquidityPool
    ✓ pauses trading and executes epoch
    ✓ deploys the hedging reactor
    ✓ can compute portfolio delta
    ✓ LP Writes a ETH/USD call for premium
    ✓ LP writes another ETH/USD call that expires later
    ✓ can compute portfolio delta
    ✓ reverts when non-admin calls rebalance function
    ✓ hedges negative delta in hedging reactor
    ✓ Adds additional liquidity from new account
    ✓ pauses trading and executes epoch
    ✓ initiates withdraw liquidity
    ✓ pauses trading and executes epoch
    ✓ LP can redeem shares
    ✓ settles an expired ITM vault
    ✓ settles an expired OTM vault
    ✓ Succeed: Hedging reactor unwind

  Options protocol
    ✓ Deploys the Option Registry
    ✓ Creates a USDC collataralised call option token series
    ✓ Reverts: Tries to close oToken series that doesnt have a vault
    ✓ Returns correct oToken when calling getOrDeployOtoken
    ✓ Returns correct oToken when calling getOToken
    ✓ Returns zero addy if option doesnt exist
    ✓ Creates a ETH collataralised call option token series
    ✓ opens call option token with USDC
    ✓ opens call option token with ETH
    ✓ writer transfers part of balance to new account
    ✓ receiver attempts to close and transaction should revert
    ✓ opens call option again with USDC
    ✓ opens call option again with ETH
    ✓ liquidityPool close and transaction succeeds
    ✓ reverts liquidityPool because of non-existent series
    ✓ liquidityPool close and transaction succeeds ETH options
    ✓ should not allow anyone outside liquidityPool to open
    ✓ Fails to settle early
    ✓ Fails to redeem early
    ✓ Fails to settle non-existent option
    ✓ Fails to redeem non-existent option
    ✓ #fastforwards time and sets oracle price
    ✓ Fails to create a USDC collataralised call option token series when expired
    ✓ Fails to open a USDC collataralised call option token series when expired
    ✓ Fails to close a USDC collataralised call option token series when expired
    ✓ settles when option expires ITM USD collateral
    ✓ reverts when attempt to settle again
    ✓ settles when option expires ITM ETH collateral
    ✓ writer redeems when option expires ITM USD collateral
    ✓ Fails when writer redeems twice
    ✓ writer redeems when option expires ITM ETH collateral
    ✓ creates a USDC collateralised call option token series
    ✓ creates a ETH collateralised call option token series
    ✓ creates a USDC put option token series
    ✓ creates a ETH put option token series
    ✓ opens put option token position
    ✓ opens an ERC20 call option
    ✓ writer transfers part of erc20 call balance to new account
    ✓ writer closes not transfered balance on ERC20 call option
    ✓ writer transfers part of put balance to new account
    ✓ writer closes not transfered balance on put option token
    ✓ settles call when option expires OTM
    ✓ writer redeems call when option expires OTM
    ✓ settles put when option expires ITM
    ✓ writer redeems put when option expires ITM
    ✓ sets the health threshold
    ✓ gets the series via issuance hash
    ✓ gets the series via series

  Options protocol Vault Health
    ✓ Deploys the Option Registry
    ✓ Creates a liquidity pool
    ✓ Creates a USDC collataralised call option token series
    ✓ Creates a ETH collataralised call option token series
    ✓ opens call option token with USDC
    ✓ opens call option token with ETH
    ✓ opens call option again with USDC
    ✓ opens call option again with ETH
    ✓ liquidityPool close and transaction succeeds
    ✓ liquidityPool close and transaction succeeds ETH options
    ✓ moves the price and changes vault health USD
    ✓ liquidityPool close and transaction succeeds
    ✓ moves the price and changes vault health ETH
    ✓ moves the price and changes vault health USD to negative rebalance stage
    ✓ readjusts to negative and checks liquidate
    ✓ reverts if unauthorised party tries to adjust collateral
    ✓ adjusts collateral to get back to positive
    ✓ readjusts to negative and checks liquidate for caller adjust
    ✓ adjusts collateral caller to get back to positive
    ✓ reverts when trying to adjust a healthy vault
    ✓ reverts adjustCollateralCaller when trying to adjust a healthy vault
    ✓ moves the price and changes vault health ETH to negative rebalance stage
    ✓ moves the price and changes vault health USD to positive rebalance stage
    ✓ adjusts overcollateralised position
    ✓ moves the price and changes vault health ETH to positive rebalance stage
    ✓ settles when option expires ITM USD collateral
    ✓ settles when option expires ITM ETH collateral
    ✓ writer redeems when option expires ITM USD collateral
    ✓ writer redeems when option expires ITM ETH collateral
    ✓ creates a USDC put option token series
    ✓ creates a ETH put option token series
    ✓ opens put option token position
    ✓ moves the price and changes vault health USD
    ✓ moves the price and changes vault health USD to negative rebalance stage
    ✓ moves the price and changes vault health USD to positive rebalance stage
    ✓ writer closes not transfered balance on put option token
    ✓ settles put when option expires ITM
    ✓ writer redeems put when option expires ITM
    ✓ Creates a USD collataralised call option token series
    ✓ moves the price and changes vault health USD to negative rebalance stage
    ✓ vault gets liquidated
    ✓ Creates a USD collataralised call option token series
    ✓ moves the price and changes vault health USD to negative rebalance stage
    ✓ vault gets partially liquidated
    ✓ vault liquidated remaining amount
    ✓ Creates a USD collataralised call option token series
    ✓ moves the price and changes vault health USD to negative rebalance stage
    ✓ vault gets liquidated by non-holder

  Oracle core logic
{
  utilizationBefore: 0,
  utilizationAfter: 0.06978290000000001,
  utilizationPrice: 474.94684003532393
}
    ✓ Sets state with written options
{
  utilizationBefore: 0,
  utilizationAfter: 0.07424309080144204,
  utilizationPrice: 474.94676488972163
}
{
  utilizationBefore: 0.06867564486699756,
  utilizationAfter: 0.14331634862171702,
  utilizationPrice: 477.96820601149875
}
    ✓ Computes portfolio delta after writing a call with intial put option from the pool
{
  utilizationBefore: 0,
  utilizationAfter: 0.07923618642779971,
  utilizationPrice: 474.9466897441059
}
{
  utilizationBefore: 0.06867564486699756,
  utilizationAfter: 0.21795705237643648,
  utilizationPrice: 955.9362551203135
}
    ✓ Computes portfolio delta after writing an additional call from an existing pool
{
  utilizationBefore: 0,
  utilizationAfter: 0.07912894348351623,
  utilizationPrice: 474.94653945283676
}
{
  utilizationBefore: 0.06869750236185022,
  utilizationAfter: 0.21653311561761357,
  utilizationPrice: 946.3765866813983
}
    ✓ Computes portfolio delta after partial buyback of option
{
  utilizationBefore: 0,
  utilizationAfter: 0.07912894348351623,
  utilizationPrice: 474.94593828725465
}
{
  utilizationBefore: 0.08061531722102894,
  utilizationAfter: 0.08061531722102894,
  utilizationPrice: 0
}
    ✓ properly computed portfolio delta after liquidation event
{
  utilizationBefore: 0,
  utilizationAfter: 0.07912894348351623,
  utilizationPrice: 0
}
{
  utilizationBefore: 0.08061531722102894,
  utilizationAfter: 0.08061531722102894,
  utilizationPrice: 0
}
    ✓ properly computes calls and puts values with expired OTM options
{
  utilizationBefore: 0,
  utilizationAfter: 0.08077560651822101,
  utilizationPrice: 0
}
{
  utilizationBefore: 0.08232509534591148,
  utilizationAfter: 0.08232509534591148,
  utilizationPrice: 0
}
{
  utilizationBefore: 1.0000000005515222,
  utilizationAfter: 1.0000000005217013,
  utilizationPrice: 2228.790767353186
}
{
  utilizationBefore: 0,
  utilizationAfter: 0.07679479463418397,
  utilizationPrice: 0
}
{
  utilizationBefore: 0.07819399619519508,
  utilizationAfter: 0.07819399619519508,
  utilizationPrice: 0
}
    ✓ properly computes portfolio value with expired ITM options
  PerpHedgingReactor
    ✓ #deploys dummy LP
    ✓ #funds accounts
    ✓ #deploy price feed
    ✓ #deploys rage
    ✓ #deploys the hedging reactor
    ✓ #deploy range order
    ✓ sets reactor address on LP contract
    ✓ returns 0 if getPoolDenominatedValue if not initialised
    ✓ reverts hedgeDelta if not initialised
    ✓ reverts update if not initialised
    ✓ initialises the reactor
[
  BigNumber { value: "1" },
  BigNumber { value: "0" },
  marketValue: BigNumber { value: "1" },
  requiredMargin: BigNumber { value: "0" }
]
BigNumber { value: "0" }
[
  BigNumber { value: "24494772735" },
  BigNumber { value: "7999361222" },
  marketValue: BigNumber { value: "24494772735" },
  requiredMargin: BigNumber { value: "7999361222" }
]
BigNumber { value: "-505227265" }
    ✓ hedges a positive delta when position is zero
[
  BigNumber { value: "24494772735" },
  BigNumber { value: "7999361222" },
  marketValue: BigNumber { value: "24494772735" },
  requiredMargin: BigNumber { value: "7999361222" }
]
BigNumber { value: "-505227265" }
[
  BigNumber { value: "1" },
  BigNumber { value: "0" },
  marketValue: BigNumber { value: "1" },
  requiredMargin: BigNumber { value: "0" }
]
BigNumber { value: "0" }
    ✓ hedges delta back to 0
[
  BigNumber { value: "1" },
  BigNumber { value: "0" },
  marketValue: BigNumber { value: "1" },
  requiredMargin: BigNumber { value: "0" }
]
BigNumber { value: "0" }
[
  BigNumber { value: "24494772735" },
  BigNumber { value: "7999361222" },
  marketValue: BigNumber { value: "24494772735" },
  requiredMargin: BigNumber { value: "7999361222" }
]
BigNumber { value: "-505227265" }
    ✓ hedges a positive delta when position is zero again
[
  BigNumber { value: "24494772735" },
  BigNumber { value: "7999361222" },
  marketValue: BigNumber { value: "24494772735" },
  requiredMargin: BigNumber { value: "7999361222" }
]
BigNumber { value: "-505227265" }
    ✓ syncs profits
    ✓ SUCCEEDS: checkvault health if price goes up
    ✓ SUCCEEDS: syncAndUpdate to get vault back on
    ✓ SUCCEEDS: checkvault health if price goes down
    ✓ SUCCEEDS: syncAndUpdate to get vault back onto normal
[
  BigNumber { value: "24999802220" },
  BigNumber { value: "7812795756" },
  marketValue: BigNumber { value: "24999802220" },
  requiredMargin: BigNumber { value: "7812795756" }
]
BigNumber { value: "-197780" }
[
  BigNumber { value: "24373346601" },
  BigNumber { value: "7617475862" },
  marketValue: BigNumber { value: "24373346601" },
  requiredMargin: BigNumber { value: "7617475862" }
]
BigNumber { value: "-1653399" }
    ✓ hedges a negative delta
    ✓ getDelta returns correct value
    ✓ gets the portfolio value
[
  BigNumber { value: "24373346601" },
  BigNumber { value: "7617475862" },
  marketValue: BigNumber { value: "24373346601" },
  requiredMargin: BigNumber { value: "7617475862" }
]
BigNumber { value: "-1653399" }
[
  BigNumber { value: "24724750842" },
  BigNumber { value: "7739309760" },
  marketValue: BigNumber { value: "24724750842" },
  requiredMargin: BigNumber { value: "7739309760" }
]
BigNumber { value: "-25249158" }
    ✓ hedges a positive delta with sufficient funds
    ✓ hedges a positive delta with insufficient funds
```

```
[
  BigNumber { value: "24736161835" },
  BigNumber { value: "7736988431" },
  marketValue: BigNumber { value: "24736161835" },
  requiredMargin: BigNumber { value: "7736988431" }
]
    ✓ liquidates usdc held position
[
  BigNumber { value: "24736064009" },
  BigNumber { value: "7736988431" },
  marketValue: BigNumber { value: "24736064009" },
  requiredMargin: BigNumber { value: "7736988431" }
]
BigNumber { value: "-13935991" }
    ✓ syncs profits
[
  BigNumber { value: "24750000000" },
  BigNumber { value: "7736988431" },
  marketValue: BigNumber { value: "24750000000" },
  requiredMargin: BigNumber { value: "7736988431" }
]
    ✓ liquidates a bit of position and withdraws sufficient funds
    ✓ update fixes balances one way
    ✓ update fixes balances other way
    ✓ update returns 0
    ✓ update reverts when not called by keeper
    ✓ liquidates all positions and withdraws
    ✓ updates healthFactor
    ✓ update health factor reverts if not owner
    ✓ withdraw reverts if not called form liquidity pool
    ✓ hedgeDelta reverts if not called from liquidity pool

  PerpHedgingReactor Sc1
    ✓ #deploys dummy LP
    ✓ #funds accounts
    ✓ #deploy price feed
    ✓ #deploys rage
    ✓ #deploys the hedging reactor
    ✓ #deploy range order
    ✓ sets reactor address on LP contract
    ✓ initialises the reactor
[
  BigNumber { value: "1" },
  BigNumber { value: "0" },
  marketValue: BigNumber { value: "1" },
  requiredMargin: BigNumber { value: "0" }
]
BigNumber { value: "0" }
[
  BigNumber { value: "24476179118" },
  BigNumber { value: "7999361222" },
  marketValue: BigNumber { value: "24476179118" },
  requiredMargin: BigNumber { value: "7999361222" }
]
BigNumber { value: "-523820882" }
    ✓ hedges a negative delta when position is zero
    ✓ SUCCEEDS: checkvault health if price goes up
    ✓ SUCCEEDS: syncAndUpdate to get vault back on
    ✓ SUCCEEDS: checkvault health if price goes down
    ✓ SUCCEEDS: syncAndUpdate to get vault back onto normal
[
  BigNumber { value: "25000171363" },
  BigNumber { value: "8192839133" },
  marketValue: BigNumber { value: "25000171363" },
  requiredMargin: BigNumber { value: "8192839133" }
]
BigNumber { value: "171363" }
[
  BigNumber { value: "25623878614" },
  BigNumber { value: "8397660112" },
  marketValue: BigNumber { value: "25623878614" },
  requiredMargin: BigNumber { value: "8397660112" }
]
BigNumber { value: "-1121386" }
    ✓ hedges more negative delta
[
  BigNumber { value: "25623878614" },
  BigNumber { value: "8397660112" },
  marketValue: BigNumber { value: "25623878614" },
  requiredMargin: BigNumber { value: "8397660112" }
]
BigNumber { value: "-1121386" }
    ✓ syncs profits
[
  BigNumber { value: "25625000000" },
  BigNumber { value: "8402699968" },
  marketValue: BigNumber { value: "25625000000" },
  requiredMargin: BigNumber { value: "8402699968" }
]
BigNumber { value: "0" }
[
  BigNumber { value: "24998802532" },
  BigNumber { value: "8197756066" },
  marketValue: BigNumber { value: "24998802532" },
  requiredMargin: BigNumber { value: "8197756066" }
]
BigNumber { value: "-1197468" }
    ✓ hedges a positive delta
    ✓ getDelta returns correct value
    ✓ gets the portfolio value
[
  BigNumber { value: "24998802532" },
  BigNumber { value: "8197756066" },
  marketValue: BigNumber { value: "24998802532" },
  requiredMargin: BigNumber { value: "8197756066" }
]
BigNumber { value: "-1197468" }
[
  BigNumber { value: "25348550362" },
  BigNumber { value: "8315731720" },
  marketValue: BigNumber { value: "25348550362" },
  requiredMargin: BigNumber { value: "8315731720" }
]
BigNumber { value: "-26449638" }
    ✓ hedges a negative delta with sufficient funds
    ✓ hedges a negative delta with insufficient funds
[
  BigNumber { value: "25365271163" },
  BigNumber { value: "8319058512" },
  marketValue: BigNumber { value: "25365271163" },
  requiredMargin: BigNumber { value: "8319058512" }
]
    ✓ liquidates a bit of position and withdraws sufficient funds
    ✓ update fixes balances one way
    ✓ update fixes balances other way
    ✓ update returns 0
    ✓ liquidates all positions and withdraws

  PerpHedgingReactor Sc2
    ✓ #deploys dummy LP
    ✓ #funds accounts
    ✓ #deploy price feed
    ✓ #deploys rage
    ✓ #deploys the hedging reactor
    ✓ #deploy range order
    ✓ sets reactor address on LP contract
    ✓ initialises the reactor
[
  BigNumber { value: "1" },
  BigNumber { value: "0" },
  marketValue: BigNumber { value: "1" },
  requiredMargin: BigNumber { value: "0" }
]
BigNumber { value: "0" }
[
  BigNumber { value: "24476179118" },
  BigNumber { value: "7999361222" },
  marketValue: BigNumber { value: "24476179118" },
  requiredMargin: BigNumber { value: "7999361222" }
]
BigNumber { value: "-523820882" }
    ✓ hedges a negative delta when position is zero
[
  BigNumber { value: "25443823275" },
  BigNumber { value: "8192839133" },
  marketValue: BigNumber { value: "25443823275" },
  requiredMargin: BigNumber { value: "8192839133" }
]
BigNumber { value: "443823275" }
[
  BigNumber { value: "29142572144" },
  BigNumber { value: "8397660112" },
  marketValue: BigNumber { value: "29142572144" },
  requiredMargin: BigNumber { value: "8397660112" }
]
BigNumber { value: "442572144" }
    ✓ hedges more negative delta
[
  BigNumber { value: "29168056665" },
  BigNumber { value: "8402699968" },
  marketValue: BigNumber { value: "29168056665" },
  requiredMargin: BigNumber { value: "8402699968" }
]
BigNumber { value: "468056665" }
[
  BigNumber { value: "29466914181" },
  BigNumber { value: "8197756066" },
  marketValue: BigNumber { value: "29466914181" },
  requiredMargin: BigNumber { value: "8197756066" }
]
BigNumber { value: "466914181" }
    ✓ hedges a positive delta
[
  BigNumber { value: "29466914181" },
  BigNumber { value: "8197756066" },
  marketValue: BigNumber { value: "29466914181" },
  requiredMargin: BigNumber { value: "8197756066" }
]
BigNumber { value: "466914181" }
    ✓ syncs profits
    ✓ getDelta returns correct value
    ✓ gets the portfolio value
[
  BigNumber { value: "29000312006" },
  BigNumber { value: "8192839133" },
  marketValue: BigNumber { value: "29000312006" },
  requiredMargin: BigNumber { value: "8192839133" }
]
BigNumber { value: "312006" }
[
  BigNumber { value: "30449709240" },
  BigNumber { value: "8315731720" },
  marketValue: BigNumber { value: "30449709240" },
  requiredMargin: BigNumber { value: "8315731720" }
```

```
]
BigNumber { value: "-290760" }
    ✓ hedges a negative delta with sufficient funds
    ✓ hedges a negative delta with insufficient funds
[
  BigNumber { value: "30466495805" },
  BigNumber { value: "8319058512" },
  marketValue: BigNumber { value: "30466495805" },
  requiredMargin: BigNumber { value: "8319058512" }
]
BigNumber { value: "16495805" }
    ✓ syncs profits
[
  BigNumber { value: "30450347643" },
  BigNumber { value: "8319058512" },
  marketValue: BigNumber { value: "30450347643" },
  requiredMargin: BigNumber { value: "8319058512" }
]
    ✓ liquidates a bit of position and withdraws sufficient funds
[
  BigNumber { value: "30450408925" },
  BigNumber { value: "8319058512" },
  marketValue: BigNumber { value: "30450408925" },
  requiredMargin: BigNumber { value: "8319058512" }
]
BigNumber { value: "408925" }
    ✓ update fixes balances one way
    ✓ update fixes balances other way
    ✓ update returns 0
    ✓ liquidates all positions and withdraws

  APVF gas tests
    ✓ SETUP: make all settings lenient
    Spin up a bunch of options and try a fulfill
      ✓ SETUP: Spin up a bunch of options
      ✓ SUCCEEDS: Calls fulfill on the options
    Try a migration with all the options
      ✓ SETUP: Make a new portfolio values feed
      ✓ SUCCEEDS: Tries to migrate to a new portfolio values feed
      ✓ SUCCEEDS: Checks the new fulfill are the same as the old fulfill
      ✓ SETUP: reconfigure original portfolio values feed
    Expire some of the options and try a clean
      ✓ SETUP: fastforward 3 days so options have expired
      ✓ SUCCEEDS: Cleans one expired option manually
      ✓ FAILS: Cleans one expired option manually with incorrect address
      ✓ FAILS: Cleans one option that is not expired
      ✓ SUCCEEDS: Cleans all expired options
    Expire some of the options at the end and try a clean
      ✓ SETUP: writes some options at the end of the array that expire soon
      ✓ SETUP: increments option series already stored
      ✓ SETUP: fastforward 3 days so options have expired
      ✓ SUCCEEDS: Cleans all expired options
    Expire some of the options and try a fulfill without first cleaning
      ✓ SETUP: fastforward 3 days so options have expired
      ✓ FAILS: Fulfill fails because of expired options not cleaned
      ✓ SUCCEEDS: Cleans all expired options
      ✓ SUCCEEDS: Fulfills correctly
    Reduce the short exposure on a series and check fulfill
      ✓ SUCCEEDS: reduces the short exposure on a series and checks the fulfill
    Add long exposure and check fulfill
      ✓ SUCCEEDS: increases the long exposure on a series and checks the fulfill
      ✓ SETUP: removes all short from index 10
      ✓ REVERTS: cant account liquidated series with no short
      ✓ REVERTS: cant account with no vault
    Access Control checks
      ✓ SUCCEEDS: set liquidity pool
      ✓ FAILS: set liquidity pool when not approved
      ✓ SUCCEEDS: set protocol
      ✓ FAILS: set protocol when not approved
      ✓ SUCCEEDS: set rfr
      ✓ FAILS: set rfr when not approved
      ✓ SUCCEEDS: set keeper
      ✓ SUCCEEDS: remove keeper
      ✓ FAILS: set keeper when not approved
      ✓ SUCCEEDS: set handler
      ✓ SUCCEEDS: remove handler
      ✓ FAILS: set keeper when not approved
      ✓ FAILS: update stores if not handler
      ✓ FAILS: sync looper if not handler
      ✓ FAILS: clean looper manually if not handler
      ✓ FAILS: migration if not governance

  Price Feed
    ✓ Should deploy price feed
    ✓ Should return a price quote
    ✓ Should return a normalized price quote
    ✓ Should return a normalised price quote on e18 decimals
    ✓ Should revert for a non-existent price quote
    ✓ Should revert for a non-existent normalised price quote

  UniswapV3HedgingReactor
    ✓ deploys the dummy LP contract
    ✓ funds the LP contract with a million USDC
    ✓ Should deploy price feed
    ✓ deploys the hedging reactor
    ✓ updates minAmount parameter
    ✓ sets reactor address on LP contract
    ✓ changes nothing if no ETH balance and hedging positive delta
    ✓ hedges a negative delta
    ✓ getDelta returns correct value
    ✓ gets the portfolio value
    ✓ hedges a positive delta with sufficient funds
```

```
    ✓ hedges a positive delta with insufficient funds
    ✓ withdraws funds without liquidation
    ✓ liquidates WETH and withdraws sufficient funds
    ✓ liquidates all ETH and withdraws but does not have enough funds
    ✓ update changes no balances
    ✓ updates poolFee
    ✓ update pool fee reverts if not owner
    ✓ withdraw reverts if not called form liquidity pool
    ✓ hedgeDelta reverts if not called from liquidity pool

·---------------------------------------------------------------------------------------------------------
|                                                    Solc version: 0.6.8
··········································································································
|  Methods
··········································································································|···············
|  Contract                                                                               ·  Method
··········································································································|···············
|  @openzeppelin/contracts-upgradeable/token/ERC20/ERC20Upgradeable.sol:ERC20Upgradeable  ·  approve
··········································································································|···············
|  @openzeppelin/contracts-upgradeable/token/ERC20/ERC20Upgradeable.sol:ERC20Upgradeable  ·  transfer
··········································································································|···············
|  AddressBook                                                                            ·  setController
··········································································································|···············
|  AddressBook                                                                            ·  setMarginCalc
··········································································································|···············
|  AddressBook                                                                            ·  setWhitelist
··········································································································|···············
|  AlphaOptionHandler                                                                     ·  createOrder
··········································································································|···············
|  AlphaOptionHandler                                                                     ·  createStrangl
··········································································································|···············
|  AlphaOptionHandler                                                                     ·  executeBuyBac
··········································································································|···············
|  AlphaOptionHandler                                                                     ·  executeOrder
··········································································································|···············
|  AlphaOptionHandler                                                                     ·  executeStrang
··········································································································|···············
|  AlphaOptionHandler                                                                     ·  setCustomOrde
··········································································································|···············
|  AlphaPortfolioValuesFeed                                                               ·  accountLiquid
··········································································································|···············
|  AlphaPortfolioValuesFeed                                                               ·  cleanLooperMa
··········································································································|···············
|  AlphaPortfolioValuesFeed                                                               ·  fulfill
··········································································································|···············
|  AlphaPortfolioValuesFeed                                                               ·  migrate
··········································································································|···············
|  AlphaPortfolioValuesFeed                                                               ·  setHandler
··········································································································|···············
|  AlphaPortfolioValuesFeed                                                               ·  setKeeper
··········································································································|···············
```

```
|  AlphaPortfolioValuesFeed                                                          ·  setLiquidityP
·····························································································|···············
|  AlphaPortfolioValuesFeed                                                          ·  setProtocol
·····························································································|···············
|  AlphaPortfolioValuesFeed                                                          ·  setRFR
·····························································································|···············
|  AlphaPortfolioValuesFeed                                                          ·  syncLooper
·····························································································|···············
|  AlphaPortfolioValuesFeed                                                          ·  updateStores
·····························································································|···············
|  Authority                                                                         ·  pullGovernor
·····························································································|···············
|  Authority                                                                         ·  pullManager
·····························································································|···············
|  Authority                                                                         ·  pushGovernor
·····························································································|···············
|  Authority                                                                         ·  pushGuardian
·····························································································|···············
|  Authority                                                                         ·  pushManager
·····························································································|···············
|  Authority                                                                         ·  revokeGuardia
·····························································································|···············
|  ClearingHouse                                                                     ·  createAccount
·····························································································|···············
|  ClearingHouse                                                                     ·  updateCollate
·····························································································|···············
|  ClearingHouse                                                                     ·  updateMargin
·····························································································|···············
|  ClearingHouse                                                                     ·  updateProtoco
·····························································································|···············
|  ClearingHouse                                                                     ·  updateRangeOr
·····························································································|···············
|  contracts/Protocol.sol:Protocol                                                   ·  changeAccount
·····························································································|···············
|  contracts/Protocol.sol:Protocol                                                   ·  changePortfol
·····························································································|···············
|  contracts/Protocol.sol:Protocol                                                   ·  changePriceFe
·····························································································|···············
|  contracts/Protocol.sol:Protocol                                                   ·  changeVolatil
·····························································································|···············
|  Controller                                                                        ·  operate
·····························································································|···············
|  Controller                                                                        ·  refreshConfig
·····························································································|···············
|  Controller                                                                        ·  setNakedCap
·····························································································|···············
|  ForceSend                                                                         ·  go
·····························································································|···············
|  LiquidityPool                                                                     ·  changeHandler
·····························································································|···············
|  LiquidityPool                                                                     ·  completeWithd
```

```
·········································································|··············
|  LiquidityPool                                                         ·  deposit
·········································································|··············
|  LiquidityPool                                                         ·  executeEpochC
·········································································|··············
|  LiquidityPool                                                         ·  initiateWithd
·········································································|··············
|  LiquidityPool                                                         ·  pause
·········································································|··············
|  LiquidityPool                                                         ·  pauseTradingA
·········································································|··············
|  LiquidityPool                                                         ·  pauseUnpauseT
·········································································|··············
|  LiquidityPool                                                         ·  rebalancePort
·········································································|··············
|  LiquidityPool                                                         ·  redeem
·········································································|··············
|  LiquidityPool                                                         ·  removeHedging
·········································································|··············
|  LiquidityPool                                                         ·  setBidAskSpre
·········································································|··············
|  LiquidityPool                                                         ·  setBufferPerc
·········································································|··············
|  LiquidityPool                                                         ·  setCollateral
·········································································|··············
|  LiquidityPool                                                         ·  setHedgingRea
·········································································|··············
|  LiquidityPool                                                         ·  setKeeper
·········································································|··············
|  LiquidityPool                                                         ·  setMaxDiscoun
·········································································|··············
|  LiquidityPool                                                         ·  setMaxPriceDe
·········································································|··············
|  LiquidityPool                                                         ·  setMaxTimeDev
·········································································|··············
|  LiquidityPool                                                         ·  setNewOptionP
·········································································|··············
|  LiquidityPool                                                         ·  setRiskFreeRa
·········································································|··············
|  LiquidityPool                                                         ·  settleVault
·········································································|··············
|  LiquidityPool                                                         ·  setUtilizatio
·········································································|··············
|  LiquidityPool                                                         ·  unpause
·········································································|··············
|  LiquidityPoolAdjustCollateralTest                                     ·  setCollateral
·········································································|··············
|  LiquidityPoolAdjustCollateralTest                                     ·  settle
·········································································|··············
|  MockChainlinkAggregator                                               ·  setLatestAnsw
·········································································|··············
```

```
|  MockChainlinkAggregator                                                              ·  setRoundAnswe
·······································································································|···············
|  MockChainlinkAggregator                                                              ·  setRoundTimes
·······································································································|···············
|  MockPortfolioValuesFeed                                                              ·  fulfill
·······································································································|···············
|  MockPortfolioValuesFeed                                                              ·  setAddressStr
·······································································································|···············
|  MockPortfolioValuesFeed                                                              ·  setKeeper
·······································································································|···············
|  MockPortfolioValuesFeed                                                              ·  setLiquidityP
·······································································································|···············
|  NewController                                                                        ·  initialize
·······································································································|···············
|  NewMarginCalculator                                                                  ·  setSpotShock
·······································································································|···············
|  NewMarginCalculator                                                                  ·  setUpperBound
·······································································································|···············
|  NewWhitelist                                                                         ·  whitelistColl
·······································································································|···············
|  NewWhitelist                                                                         ·  whitelistCove
·······································································································|···············
|  NewWhitelist                                                                         ·  whitelistNake
·······································································································|···············
|  NewWhitelist                                                                         ·  whitelistProd
·······································································································|···············
|  OptionHandler                                                                        ·  addOrRemoveBu
·······································································································|···············
|  OptionHandler                                                                        ·  buybackOption
·······································································································|···············
|  OptionHandler                                                                        ·  createOrder
·······································································································|···············
|  OptionHandler                                                                        ·  createStrangl
·······································································································|···············
|  OptionHandler                                                                        ·  executeOrder
·······································································································|···············
|  OptionHandler                                                                        ·  executeStrang
·······································································································|···············
|  OptionHandler                                                                        ·  issue
·······································································································|···············
|  OptionHandler                                                                        ·  issueAndWrite
·······································································································|···············
|  OptionHandler                                                                        ·  pause
·······································································································|···············
|  OptionHandler                                                                        ·  setCustomOrde
·······································································································|···············
|  OptionHandler                                                                        ·  unpause
·······································································································|···············
|  OptionHandler                                                                        ·  writeOption
·······································································································|···············
|  OptionRegistry                                                                       ·  adjustCollate
```

```
..........................................................................................|...............
|  OptionRegistry                                                                         ·  adjustCollate
..........................................................................................|...............
|  OptionRegistry                                                                         ·  close
..........................................................................................|...............
|  OptionRegistry                                                                         ·  issue
..........................................................................................|...............
|  OptionRegistry                                                                         ·  open
..........................................................................................|...............
|  OptionRegistry                                                                         ·  redeem
..........................................................................................|...............
|  OptionRegistry                                                                         ·  registerLiqui
..........................................................................................|...............
|  OptionRegistry                                                                         ·  setHealthThre
..........................................................................................|...............
|  OptionRegistry                                                                         ·  setKeeper
..........................................................................................|...............
|  OptionRegistry                                                                         ·  setLiquidityP
..........................................................................................|...............
|  OptionRegistry                                                                         ·  settle
..........................................................................................|...............
|  OptionRegistry                                                                         ·  wCollatLiquid
..........................................................................................|...............
|  Oracle                                                                                 ·  setAssetPrice
..........................................................................................|...............
|  Oracle                                                                                 ·  setExpiryPric
..........................................................................................|...............
|  Oracle                                                                                 ·  setStablePric
..........................................................................................|...............
|  OracleMock                                                                             ·  setSqrtPriceX
..........................................................................................|...............
|  PerpHedgingReactor                                                                     ·  initialiseRea
..........................................................................................|...............
|  PerpHedgingReactor                                                                     ·  setHealthFact
..........................................................................................|...............
|  PerpHedgingReactor                                                                     ·  setKeeper
..........................................................................................|...............
|  PerpHedgingReactor                                                                     ·  syncAndUpdate
..........................................................................................|...............
|  PerpHedgingTest                                                                        ·  hedgeDelta
..........................................................................................|...............
|  PerpHedgingTest                                                                        ·  setHedgingRea
..........................................................................................|...............
|  PerpHedgingTest                                                                        ·  syncAndUpdate
..........................................................................................|...............
|  PerpHedgingTest                                                                        ·  update
..........................................................................................|...............
|  PerpHedgingTest                                                                        ·  withdraw
..........................................................................................|...............
|  PriceFeed                                                                              ·  addPriceFeed
..........................................................................................|...............
```

```
|  RageTradeFactory                                                                       ·  initializePoo
..........................................................................................|...............
|  UniswapV3HedgingReactor                                                                ·  changePoolFee
..........................................................................................|...............
|  UniswapV3HedgingReactor                                                                ·  setMinAmount
..........................................................................................|...............
|  UniswapV3HedgingReactor                                                                ·  setSlippage
..........................................................................................|...............
|  UniswapV3HedgingTest                                                                   ·  hedgeDelta
..........................................................................................|...............
|  UniswapV3HedgingTest                                                                   ·  setHedgingRea
..........................................................................................|...............
|  UniswapV3HedgingTest                                                                   ·  update
..........................................................................................|...............
|  UniswapV3HedgingTest                                                                   ·  withdraw
..........................................................................................|...............
|  VolatilityFeed                                                                         ·  setVolatility
..........................................................................................|...............
|  WETH9                                                                                  ·  deposit
..........................................................................................|...............
|  Deployments
..........................................................................................................
|  Account
..........................................................................................................
|  Accounting
..........................................................................................................
|  AlphaOptionHandler
..........................................................................................................
|  AlphaPortfolioValuesFeed
..........................................................................................................
|  Authority
..........................................................................................................
|  BlackScholes
..........................................................................................................
|  BlackScholesTest
..........................................................................................................
|  ChainLinkPricer
..........................................................................................................
|  ClearingHouse
..........................................................................................................
|  ClearingHouseLens
..........................................................................................................
|  contracts/Protocol.sol:Protocol
..........................................................................................................
|  ForceSend
..........................................................................................................
|  InsuranceFund
..........................................................................................................
|  LiquidityPool
..........................................................................................................
|  LiquidityPoolAdjustCollateralTest
```

```
·············································································································
|  MarginVault
·············································································································
|  MockChainlinkAggregator
·············································································································
|  MockPortfolioValuesFeed
·············································································································
|  NewController
·············································································································
|  NewMarginCalculator
·············································································································
|  NewWhitelist
·············································································································
|  NormalDist
·············································································································
|  OptionHandler
·············································································································
|  OptionRegistry
·············································································································
|  OptionsCompute
·············································································································
|  OpynInteractions
·············································································································
|  OracleMock
·············································································································
|  PerpHedgingReactor
·············································································································
|  PerpHedgingTest
·············································································································
|  PriceFeed
·············································································································
|  RageTradeFactory
·············································································································
|  SettlementTokenOracle
·············································································································
|  UniswapV3HedgingReactor
·············································································································
|  UniswapV3HedgingTest
·············································································································
|  Volatility
·············································································································
|  VolatilityFeed
·············································································································
|  VPoolWrapper
·--------------------------------------------------------------------------------------------------------

  538 passing (4m)
```

Coverage

One of the tests failed during test coverage. I spoke with the Rysk team about this and we agreed that because it's a rounding error, it is not something that needs further work to fix.

```
❯ npm run test-coverage

> delta-hedging@1.0.0 test-coverage
> export NODE_OPTIONS='--max-old-space-size=8192' && hardhat coverage --testfiles 'test/*.ts'

(node:27476) Warning: Accessing non-existent property 'INVALID_ALT_NUMBER' of module exports inside circul
(Use `node --trace-warnings ...` to show where the warning was created)
(node:27476) Warning: Accessing non-existent property 'INVALID_ALT_NUMBER' of module exports inside circul

Version
=======
> solidity-coverage: v0.7.20

Instrumenting for coverage...
=============================

> Accounting.sol
> AlphaOptionHandler.sol
> AlphaPortfolioValuesFeed.sol
> Authority.sol
> hedging/PerpHedgingReactor.sol
> hedging/UniswapV3HedgingReactor.sol
> interfaces/AddressBookInterface.sol
> interfaces/AggregatorV3Interface.sol
> interfaces/GammaInterface.sol
> interfaces/I_ERC20.sol
> interfaces/IAccounting.sol
> interfaces/IAuthority.sol
> interfaces/IHedgingReactor.sol
> interfaces/ILiquidityPool.sol
> interfaces/IMarginCalculator.sol
> interfaces/IOptionRegistry.sol
> interfaces/IOracle.sol
> interfaces/IPortfolioValuesFeed.sol
> interfaces/WETH.sol
> libraries/AccessControl.sol
> libraries/BlackScholes.sol
> libraries/CustomErrors.sol
> libraries/EnumerableSet.sol
> libraries/NormalDist.sol
> libraries/OptionsCompute.sol
> libraries/OpynInteractions.sol
> libraries/SafeTransferLib.sol
> libraries/Types.sol
> LiquidityPool.sol
> mocks/MockPortfolioValuesFeed.sol
> OptionHandler.sol
> OptionRegistry.sol
> packages/opyn/core/AddressBook.sol
> packages/opyn/core/Controller.sol
> packages/opyn/core/MarginCalculator.sol
> packages/opyn/core/MarginPool.sol
> packages/opyn/core/Oracle.sol
> packages/opyn/core/Otoken.sol
> packages/opyn/core/OtokenFactory.sol
> packages/opyn/core/OtokenSpawner.sol
> packages/opyn/core/Whitelist.sol
> packages/opyn/external/callees/PermitCallee.sol
> packages/opyn/external/canonical-weth/WETH9.sol
> packages/opyn/external/proxies/PayableProxyController.sol
> packages/opyn/interfaces/AddressBookInterface.sol
> packages/opyn/interfaces/AggregatorInterface.sol
> packages/opyn/interfaces/CalleeInterface.sol
> packages/opyn/interfaces/CTokenInterface.sol
> packages/opyn/interfaces/ERC20Interface.sol
> packages/opyn/interfaces/MarginCalculatorInterface.sol
> packages/opyn/interfaces/MarginPoolInterface.sol
> packages/opyn/interfaces/OpynPricerInterface.sol
> packages/opyn/interfaces/OracleInterface.sol
> packages/opyn/interfaces/OtokenInterface.sol
> packages/opyn/interfaces/WETH9Interface.sol
> packages/opyn/interfaces/WhitelistInterface.sol
> packages/opyn/interfaces/WSTETHInterface.sol
> packages/opyn/interfaces/YearnVaultInterface.sol
> packages/opyn/interfaces/ZeroXExchangeInterface.sol
> packages/opyn/libs/Actions.sol
> packages/opyn/libs/FixedPointInt256.sol
> packages/opyn/libs/MarginVault.sol
> packages/opyn/libs/SignedConverter.sol
> packages/opyn/Migrations.sol
> packages/opyn/mocks/Mock0xERC20Proxy.sol
> packages/opyn/mocks/Mock0xExchange.sol
> packages/opyn/mocks/MockAddressBook.sol
> packages/opyn/mocks/MockChainlinkAggregator.sol
> packages/opyn/mocks/MockController.sol
> packages/opyn/mocks/MockCToken.sol
> packages/opyn/mocks/MockCUSDC.sol
> packages/opyn/mocks/MockDumbERC20.sol
> packages/opyn/mocks/MockERC20.sol
> packages/opyn/mocks/MockOracle.sol
> packages/opyn/mocks/MockOtoken.sol
> packages/opyn/mocks/MockPermitERC20.sol
> packages/opyn/mocks/MockPricer.sol
> packages/opyn/mocks/MockWhitelistModule.sol
> packages/opyn/mocks/MockWSTETHToken.sol
> packages/opyn/mocks/MockYToken.sol
> packages/opyn/new/NewCalculator.sol
> packages/opyn/new/NewController.sol
> packages/opyn/new/NewMarginCalculatorInterface.sol
> packages/opyn/new/NewWhitelist.sol
> packages/opyn/packages/BokkyPooBahsDateTimeLibrary.sol
> packages/opyn/packages/oz/Address.sol
> packages/opyn/packages/oz/Context.sol
> packages/opyn/packages/oz/Create2.sol
> packages/opyn/packages/oz/IERC20.sol
> packages/opyn/packages/oz/Ownable.sol
> packages/opyn/packages/oz/ReentrancyGuard.sol
> packages/opyn/packages/oz/SafeERC20.sol
> packages/opyn/packages/oz/SafeMath.sol
> packages/opyn/packages/oz/SignedSafeMath.sol
> packages/opyn/packages/oz/Strings.sol
> packages/opyn/packages/oz/upgradeability/cryptography/ECDSAUpgradeable.sol
> packages/opyn/packages/oz/upgradeability/erc20-permit/EIP712Upgradeable.sol
> packages/opyn/packages/oz/upgradeability/erc20-permit/ERC20PermitUpgradeable.sol
> packages/opyn/packages/oz/upgradeability/erc20-permit/IERC20PermitUpgradeable.sol
> packages/opyn/packages/oz/upgradeability/ERC20Upgradeable.sol
> packages/opyn/packages/oz/upgradeability/GSN/ContextUpgradeable.sol
> packages/opyn/packages/oz/upgradeability/IERC20Upgradeable.sol
> packages/opyn/packages/oz/upgradeability/Initializable.sol
> packages/opyn/packages/oz/upgradeability/math/SafeMathUpgradeable.sol
> packages/opyn/packages/oz/upgradeability/OwnableUpgradeSafe.sol
> packages/opyn/packages/oz/upgradeability/OwnedUpgradeabilityProxy.sol
> packages/opyn/packages/oz/upgradeability/Proxy.sol
> packages/opyn/packages/oz/upgradeability/ReentrancyGuardUpgradeSafe.sol
> packages/opyn/packages/oz/upgradeability/UpgradeabilityProxy.sol
> packages/opyn/packages/oz/upgradeability/utils/CountersUpgradeable.sol
> packages/opyn/packages/Spawn.sol
> packages/opyn/pricers/ChainlinkPricer.sol
> packages/opyn/pricers/CompoundPricer.sol
> packages/opyn/pricers/WstethPricer.sol
> packages/opyn/pricers/YearnPricer.sol
> packages/opyn/tests/ActionTester.sol
> packages/opyn/tests/CalculatorTester.sol
> packages/opyn/tests/CalleeAllowanceTester.sol
> packages/opyn/tests/CallTester.sol
> packages/opyn/tests/FixedPointInt256Tester.sol
> packages/opyn/tests/FlashUnwrap.sol
> packages/opyn/tests/ForceSend.sol
> packages/opyn/tests/MarginVaultTester.sol
> packages/opyn/tests/OtokenImplV1.sol
> packages/opyn/tests/SignedConverterTester.sol
> packages/opyn/tests/UpgradeableContractV1.sol
> packages/opyn/tests/UpgradeableContractV2.sol
> PortfolioValuesFeed.sol
> PriceFeed.sol
> Protocol.sol
> tokens/ERC20.sol
> tokens/MintableERC20.sol
> tokens/WETH.sol
> utils/BlackScholesTest.sol
> utils/LiquidityPoolAdjustCollateralTest.sol
> utils/OracleMock.sol
> utils/PerpHedgingTest.sol
> utils/RealTokenMock.sol
> utils/ReentrancyGuard.sol
> utils/UniswapV3HedgingTest.sol
> utils/Volatility.sol
> VolatilityFeed.sol

Compilation:
============

Warning: Source file does not specify required compiler version! Consider adding "pragma solidity ^0.8.9;"
--> contracts/utils/LiquidityPoolAdjustCollateralTest.sol


Warning: Unused function parameter. Remove or comment out the variable name to silence this warning.
   --> contracts/AlphaPortfolioValuesFeed.sol:506:12:
    |
506 |           returns (bytes32 id)
    |                    ^^^^^^^^^^


Warning: Unused function parameter. Remove or comment out the variable name to silence this warning.
   --> contracts/utils/LiquidityPoolAdjustCollateralTest.sol:134:22:
    |
134 |   function getBalance(address collateralAsset) external view returns (uint256){c ...
    |                       ^^^^^^^^^^^^^^^^^^^^^^^


Warning: Contract code size exceeds 24576 bytes (a limit introduced in Spurious Dragon). This contract may
  --> contracts/AlphaOptionHandler.sol:28:1:
   |
28 | contract AlphaOptionHandler is AccessControl, ReentrancyGuard {
   | ^ (Relevant source part starts here and spans across multiple lines).


Warning: Contract code size exceeds 24576 bytes (a limit introduced in Spurious Dragon). This contract may
  --> contracts/AlphaPortfolioValuesFeed.sol:26:1:
   |
26 | contract AlphaPortfolioValuesFeed is AccessControl, IPortfolioValuesFeed {
   | ^ (Relevant source part starts here and spans across multiple lines).


Warning: Contract code size exceeds 24576 bytes (a limit introduced in Spurious Dragon). This contract may
  --> contracts/LiquidityPool.sol:33:1:
   |
33 | contract LiquidityPool is ERC20, AccessControl, ReentrancyGuard, Pausable {
   | ^ (Relevant source part starts here and spans across multiple lines).
```

```
Warning: Contract code size exceeds 24576 bytes (a limit introduced in Spurious Dragon). This contract may
  --> contracts/OptionHandler.sol:30:1:
   |
30 | contract OptionHandler is Pausable, AccessControl, ReentrancyGuard {
   | ^ (Relevant source part starts here and spans across multiple lines).


Warning: Contract code size exceeds 24576 bytes (a limit introduced in Spurious Dragon). This contract may
  --> contracts/OptionRegistry.sol:26:1:
   |
26 | contract OptionRegistry is AccessControl {
   | ^ (Relevant source part starts here and spans across multiple lines).


contracts/packages/opyn/core/Oracle.sol:516:1: Warning: This declaration shadows an existing declaration.
Price memory price = storedPrice[_asset][_expiryTimestamp];
^----------------^
contracts/packages/opyn/core/Oracle.sol:507:1: The shadowed declaration is here:
uint256 price = stablePrice[_asset];
^-----------^

contracts/packages/opyn/packages/oz/upgradeability/Proxy.sol:10:1: Warning: This contract has a payable fa
abstract contract Proxy {
^ (Relevant source part starts here and spans across multiple lines).
contracts/packages/opyn/packages/oz/upgradeability/Proxy.sol:23:5: The payable fallback function is define
    fallback() external payable {c_0xaa9018ee(0x60c6410c329845b4f9e9f3584342609c74468cdcc3fff63f8f77bbbad0
    ^ (Relevant source part starts here and spans across multiple lines).

contracts/packages/opyn/packages/oz/upgradeability/UpgradeabilityProxy.sol:12:1: Warning: This contract ha
contract UpgradeabilityProxy is Proxy {
^ (Relevant source part starts here and spans across multiple lines).
contracts/packages/opyn/packages/oz/upgradeability/Proxy.sol:23:5: The payable fallback function is define
    fallback() external payable {c_0xaa9018ee(0x60c6410c329845b4f9e9f3584342609c74468cdcc3fff63f8f77bbbad0
    ^ (Relevant source part starts here and spans across multiple lines).

contracts/packages/opyn/packages/oz/upgradeability/OwnedUpgradeabilityProxy.sol:12:1: Warning: This contra
contract OwnedUpgradeabilityProxy is UpgradeabilityProxy {
^ (Relevant source part starts here and spans across multiple lines).
contracts/packages/opyn/packages/oz/upgradeability/Proxy.sol:23:5: The payable fallback function is define
    fallback() external payable {c_0xaa9018ee(0x60c6410c329845b4f9e9f3584342609c74468cdcc3fff63f8f77bbbad0
    ^ (Relevant source part starts here and spans across multiple lines).

contracts/packages/opyn/external/proxies/PayableProxyController.sol:21:1: Warning: This contract has a pay
contract PayableProxyController is ReentrancyGuard {
^ (Relevant source part starts here and spans across multiple lines).
contracts/packages/opyn/external/proxies/PayableProxyController.sol:50:5: The payable fallback function is
    fallback() external payable {c_0x1c4ea5c5(0xf5c6e0232160d2da92ecf7313c019f6069e40a2d750edf0e8c12c7cd30
    ^ (Relevant source part starts here and spans across multiple lines).
```

```
contracts/packages/opyn/packages/oz/upgradeability/ERC20Upgradeable.sol:451:9: Warning: Unused function pa
        address from,
        ^----------^

contracts/packages/opyn/packages/oz/upgradeability/ERC20Upgradeable.sol:452:9: Warning: Unused function pa
        address to,
        ^--------^

contracts/packages/opyn/packages/oz/upgradeability/ERC20Upgradeable.sol:453:9: Warning: Unused function pa
        uint256 amount
        ^------------^

contracts/packages/opyn/packages/oz/upgradeability/erc20-permit/ERC20PermitUpgradeable.sol:54:43: Warning:
 ... ction __ERC20Permit_init_unchained(string memory name) internal initializer {c_0xc2721d9 ...
                                        ^----------------^

contracts/packages/opyn/external/callees/PermitCallee.sol:19:27: Warning: Unused function parameter. Remov
    function callFunction(address payable _sender, bytes memory _data) external over ...
                          ^---------------------^

contracts/packages/opyn/mocks/Mock0xExchange.sol:34:9: Warning: Unused function parameter. Remove or comme
        ZeroXExchangeInterface.LimitOrder memory _order,
        ^---------------------------------------------^

contracts/packages/opyn/mocks/Mock0xExchange.sol:35:9: Warning: Unused function parameter. Remove or comme
        ZeroXExchangeInterface.Signature memory _signature,
        ^------------------------------------------------^

contracts/packages/opyn/mocks/Mock0xExchange.sol:36:9: Warning: Unused function parameter. Remove or comme
        uint128 _takerTokenFillAmount
        ^---------------------------^

contracts/packages/opyn/mocks/Mock0xExchange.sol:48:9: Warning: Unused function parameter. Remove or comme
        bool _revertIfIncomplete
        ^----------------------^

contracts/packages/opyn/mocks/MockPricer.sol:45:33: Warning: Unused function parameter. Remove or comment
    function getHistoricalPrice(uint80 _roundId) external view returns (uint256, u ...
                                ^-------------^

contracts/packages/opyn/pricers/YearnPricer.sol:127:33: Warning: Unused function parameter. Remove or comm
    function getHistoricalPrice(uint80 _roundId) external view override returns (u ...
                                ^-------------^

contracts/packages/opyn/packages/oz/upgradeability/ERC20Upgradeable.sol:450:5: Warning: Function state mut
    function _beforeTokenTransfer(
    ^ (Relevant source part starts here and spans across multiple lines).

contracts/packages/opyn/core/Controller.sol:68:1: Warning: Contract code size exceeds 24576 bytes (a limit
contract Controller is Initializable, OwnableUpgradeSafe, ReentrancyGuardUpgradeSafe {
^ (Relevant source part starts here and spans across multiple lines).
```

```
contracts/packages/opyn/core/MarginCalculator.sol:20:1: Warning: Contract code size exceeds 24576 bytes (a
contract MarginCalculator is Ownable {
^ (Relevant source part starts here and spans across multiple lines).

contracts/packages/opyn/core/Otoken.sol:16:1: Warning: Contract code size exceeds 24576 bytes (a limit int
contract Otoken is ERC20PermitUpgradeable {
^ (Relevant source part starts here and spans across multiple lines).

contracts/packages/opyn/mocks/MockOtoken.sol:10:1: Warning: Contract code size exceeds 24576 bytes (a limi
contract MockOtoken is ERC20PermitUpgradeable {
^ (Relevant source part starts here and spans across multiple lines).

contracts/packages/opyn/new/NewCalculator.sol:22:1: Warning: Contract code size exceeds 24576 bytes (a lim
contract NewMarginCalculator is Ownable {
^ (Relevant source part starts here and spans across multiple lines).

contracts/packages/opyn/new/NewController.sol:68:1: Warning: Contract code size exceeds 24576 bytes (a lim
contract NewController is Initializable, OwnableUpgradeSafe, ReentrancyGuardUpgradeSafe {
^ (Relevant source part starts here and spans across multiple lines).

contracts/packages/opyn/tests/CalculatorTester.sol:10:1: Warning: Contract code size exceeds 24576 bytes (
contract CalculatorTester is MarginCalculator {
^ (Relevant source part starts here and spans across multiple lines).

contracts/packages/opyn/tests/OtokenImplV1.sol:11:1: Warning: Contract code size exceeds 24576 bytes (a li
contract OtokenImplV1 is ERC20PermitUpgradeable {
^ (Relevant source part starts here and spans across multiple lines).

Warning: Unnamed return variable can remain unassigned. Add an explicit return with value to all non-rever
   --> contracts/tokens/MintableERC20.sol:235:65:
    |
235 |  ...  uint256 amount) external returns (bool) {c_0x6e026712(0x8318fa0b8ca964999 ...
    |                                         ^^^^


Warning: Unused local variable.
   --> contracts/hedging/PerpHedgingReactor.sol:172:2:
    |
172 | (IERC20 collateral, uint256 collat) = clearingHouse.getAccountCollateralInfo(accountId, collateralId
    |  ^^^^^^^^^^^^^^^^^


Warning: Unused function parameter. Remove or comment out the variable name to silence this warning.
   --> contracts/mocks/MockPortfolioValuesFeed.sol:160:3:
    |
160 |           bytes32 _requestId,
    |           ^^^^^^^^^^^^^^^^^^


Warning: Unused function parameter. Remove or comment out the variable name to silence this warning.
```

```
   --> contracts/mocks/MockPortfolioValuesFeed.sol:218:32:
    |
218 |   function requestPortfolioData(address _underlying, address _strike)
    |                                 ^^^^^^^^^^^^^^^^^^^


Warning: Unused function parameter. Remove or comment out the variable name to silence this warning.
   --> contracts/mocks/MockPortfolioValuesFeed.sol:218:53:
    |
218 |   function requestPortfolioData(address _underlying, address _strike)
    |                                                      ^^^^^^^^^^^^^^^


Warning: Unused function parameter. Remove or comment out the variable name to silence this warning.
   --> contracts/mocks/MockPortfolioValuesFeed.sol:220:12:
    |
220 |           returns (bytes32 requestId)
    |                    ^^^^^^^^^^^^^^^^^


Warning: Function state mutability can be restricted to view
   --> contracts/mocks/MockPortfolioValuesFeed.sol:218:2:
    |
218 |   function requestPortfolioData(address _underlying, address _strike)
    |   ^ (Relevant source part starts here and spans across multiple lines).


Warning: Contract code size is 35760 bytes and exceeds 24576 bytes (a limit introduced in Spurious Dragon)
  --> @rage/core/contracts/libraries/Account.sol:27:1:
   |
27 | library Account {
   | ^ (Relevant source part starts here and spans across multiple lines).


Warning: Contract code size is 28267 bytes and exceeds 24576 bytes (a limit introduced in Spurious Dragon)
  --> @rage/core/contracts/protocol/RageTradeFactory.sol:35:1:
   |
35 | contract RageTradeFactory is
   | ^ (Relevant source part starts here and spans across multiple lines).


Warning: Contract code size is 41983 bytes and exceeds 24576 bytes (a limit introduced in Spurious Dragon)
  --> @rage/core/contracts/protocol/clearinghouse/ClearingHouse.sol:34:1:
   |
34 | contract ClearingHouse is
   | ^ (Relevant source part starts here and spans across multiple lines).


Warning: Contract code size is 33040 bytes and exceeds 24576 bytes (a limit introduced in Spurious Dragon)
  --> @rage/core/contracts/protocol/wrapper/VPoolWrapper.sol:36:1:
   |
36 | contract VPoolWrapper is IVPoolWrapper, IUniswapV3MintCallback, IUniswapV3SwapCallback, Initializable
   | ^ (Relevant source part starts here and spans across multiple lines).



Warning: Contract code size is 33960 bytes and exceeds 24576 bytes (a limit introduced in Spurious Dragon)
  --> contracts/hedging/PerpHedgingReactor.sol:26:1:
   |
26 | contract PerpHedgingReactor is IHedgingReactor, AccessControl {
   | ^ (Relevant source part starts here and spans across multiple lines).



Generating typings for: 281 artifacts in dir: types for target: ethers-v5
Successfully generated 417 typings!
Compiled 286 Solidity files successfully

Network Info
============
> HardhatEVM: v2.9.1
> network:    hardhat

No need to generate any newer typings.



  Pricing options
    ✔ Should deploy Black Scholes library (186ms)
    ✔ correctly prices in the money call with a one year time to expiration (94ms)
    ✔ correctly  prices out of the money call with one year time (48ms)
    ✔ correctly prices out of the money call with one year time high volatility (48ms)
    ✔ correctly prices in the money call with one month expiration high volatility (44ms)
    ✔ correctly prices in the money put with one year time (43ms)
    ✔ correctly prices in the money put with one year time high volatility (44ms)
    ✔ correctly prices in the money put with one month time high volatility (41ms)
    ✔ correctly prices in the money put with one month time high volatility (42ms)
    ✔ correctly prices at the money put with one month time high volatility
    ✔ correctly prices near the money put with one month time high volatility (45ms)
    ✔ correctly prices out of the money put with one month time high volatility
    ✔ correctly prices out of the money put with one month time (41ms)
    ✔ correctly computes delta of out of the money call with one month time
    ✔ correctly computes delta of out of the money put with one month time
    ✔ Estimated portfolio deltas should deviate by more than 10% compared with cached values at scale (78m

  Authority tests
    Authority push effective immediately
      ✔ SUCCEEDS: set governor
      ✔ SUCCEEDS: set manager
      ✔ SUCCEEDS: set guardian
      ✔ SUCCEEDS: revoke guardian
      ✔ FAILS: revoke guardian when not auth
      ✔ FAILS: set governor when not auth
      ✔ FAILS: set manager when not auth
      ✔ FAILS: set guardian when not auth
    Authority push and pull
      ✔ SUCCEEDS: push governor
      ✔ FAILS: rando tries to pull governor rank
      ✔ SUCCEEDS: pull governor rank
      ✔ SUCCEEDS: push manager
      ✔ FAILS: rando tries to pull manager rank
      ✔ SUCCEEDS: pull manager rank

  Dyn Quote Tests
    ✔ Deposit to the liquidityPool (623ms)
    Quote
      ✔ gets price
      ✔ Returns a quote for a ETH/USD put with utilization
      ✔ Returns a quote for ETH/USD call with utilization
      ✔ Returns a quote for a ETH/USD put to buy
      ✔ Returns a quote for ETH/USD call to buy

  Hegic Attack
    ✔ Adds liquidity to the liquidityPool (615ms)
    ✔ Attacker adds liquidity (1228ms)
    ✔ pauses trading and executes epoch (1341ms)
    ✔ LP Writes a WETH/USD put collateralized by USD for premium to the attacker (1413ms)
    ✔ attacker initiates withdraw liquidity (303ms)
    ✔ pauses trading and executes epoch (1004ms)
    ✔ attacker withdraws liquidity (413ms)

  RR oracle between update attack vector
    Sc 1. Single large option purchase and update checks
      ✔ Sc1. Adds liquidity to the liquidityPool (620ms)
      ✔ Sc1. Another adds liquidity to the liquidityPool (630ms)
      ✔ Sc1. pauses trading and executes epoch (1926ms)
{ collateralAllocatedBefore: BigNumber { value: "0" } }
{ quote: '356055.60541235485059424' }
Pool should now have delta value of: 222850258183497507200
Pool should now have an options portfolio value of (or liabilities): 356055.5695312279
NAV after issuance should be: 2000000000000000000000000
NAV after issuance is: 2000000000000000000000000
{ collateralAllocatedAfter: BigNumber { value: "1086994811040" } }
      ✔ Sc1. LP Writes a WETH/USD put collateralized by USD for premium to the attacker (1671ms)
{
  utilizationBefore: 0,
  utilizationAfter: 0.5434974055,
  utilizationPrice: 323686.8210026461
}
{
  utilizationBefore: 0,
  utilizationAfter: 0.5434974055,
  utilizationPrice: 323686.8210026461
}
{
  portfolioDelta: 222.85028854890703,
  portfolioGamma: -0.12854325965329505,
  portfolioTheta: 2580.4151214238923,
  portfolioVega: -1373.1575705254616,
  callsPutsValue: 356055.5031029107,
  bsCallsPutsValue: 323686.8210026461
}
{
  beforeNAV: BigNumber { value: "20000000000000000000000000" },
  afterNAV: BigNumber { value: "20000000361250893000000000" }
}
{ collateralAllocated: BigNumber { value: "1086994811040" } }
      ✔ Sc1. should update NAV after fulfill (293ms)
      ✔ Sc1. initiates withdraw liquidity (679ms)
{
  utilizationBefore: 0,
  utilizationAfter: 0.5434974055,
  utilizationPrice: 323686.73075094237
}
{
  utilizationBefore: 0,
  utilizationAfter: 0.5434974055,
  utilizationPrice: 323686.73075094237
}
{
  portfolioDelta: 222.8503342266067,
  portfolioGamma: -0.12854331586370668,
  portfolioTheta: 2580.41626096762,
  portfolioVega: -1373.1568609965484,
  callsPutsValue: 356055.4038260366,
  bsCallsPutsValue: 323686.73075094237
}
      ✔ Sc1. pauses trading and executes epoch (1158ms)
liabilities are now 0 because the pool isnt updated
USDC withdrawn: 1000000067700
NAV after withdraw should be: 10000000000000000000000000
NAV after withdraw is: 10000000677019634000000000
{
  utilizationBefore: 0,
  utilizationAfter: 1.086994884915652,
  utilizationPrice: 533019.6384278896
}
{
  utilizationBefore: 0,
  utilizationAfter: 1.086994884915652,
  utilizationPrice: 533019.6384278896
}
{
  portfolioDelta: 222.8503799043529,
  portfolioGamma: -0.1285433720741847,
  portfolioTheta: 2580.417400512687,
  portfolioVega: -1373.1561514672069,
  callsPutsValue: 586321.6022706785,
  bsCallsPutsValue: 323686.640499199
}
NAV after update should be: 1000000000000000000000000
NAV after update is: 769733869257321500000000
      ✔ Sc1. attacker withdraws liquidity before delta and portfolio values update (685ms)
    Sc 2. Two Seperate Single large option purchase and update checks
      ✔ Sc2. Adds liquidity to the liquidityPool (617ms)
      ✔ Sc2. Another adds liquidity to the liquidityPool (1218ms)
      ✔ Sc2. pauses trading and executes epoch (985ms)
{ collateralAllocatedBefore: BigNumber { value: "0" } }
{ quote: '356055.60541235485059424' }
Pool should now have delta value of: 222850258183497507200
Pool should now have an options portfolio value of (or liabilities): 356055.5695312279
NAV after issuance should be: 2000000000000000000000000
NAV after issuance is: 2000000000000000000000000
{ collateralAllocatedAfter: BigNumber { value: "1086994811040" } }
      ✔ Sc2. LP Writes a WETH/USD put collateralized by USD for premium to the attacker (1369ms)
{ collateralAllocatedBefore: BigNumber { value: "1086994811040" } }
{ quote: '67253.97931894967072049' }
Pool should now have delta value of: 297851664267101279300
Pool should now have an options portfolio value of (or liabilities): 423309.5248438678
NAV after issuance should be: 2000000000000000000000000
NAV after issuance is: 2000000000000000000000000
{ collateralAllocatedAfter: BigNumber { value: "1339664439167" } }
      ✔ Sc2. LP Writes a WETH/USD call collateralized by USD for premium to the attacker (1574ms)
{
  utilizationBefore: 0,
  utilizationAfter: 0.5990323996728818,
  utilizationPrice: 323686.73075094237
}
{
  utilizationBefore: 0.461362133874553,
  utilizationAfter: 0.5686047789724876,
  utilizationPrice: 74726.60409264026
}
{
  utilizationBefore: 0,
  utilizationAfter: 0.5990323996728818,
  utilizationPrice: 323686.73075094237
}
{
  utilizationBefore: 0.461362133874553,
  utilizationAfter: 0.5686047789724876,
  utilizationPrice: 74726.60409264026
}
{
  portfolioDelta: 147.84892604378842,
  portfolioGamma: -0.15385780646332958,
  portfolioTheta: 3137.4510566071394,
  portfolioVega: -1649.4322167313362,
  callsPutsValue: 406959.5432520756,
  bsCallsPutsValue: 398413.33484358265
}
{
  beforeNAV: BigNumber { value: "2000000000000000000000000" },
  afterNAV: BigNumber { value: "2016349963623924400000000" }
}
{ collateralAllocated: BigNumber { value: "1339664439167" } }
      ✔ Sc2. should update NAV after fulfill (564ms)
      ✔ Sc2. initiates withdraw liquidity (694ms)
{
  utilizationBefore: 0,
  utilizationAfter: 0.5990323996728818,
  utilizationPrice: 323686.640499199
}
{
  utilizationBefore: 0.461362133874553,
  utilizationAfter: 0.5686047789724876,
  utilizationPrice: 74726.58464112617
}
{
  utilizationBefore: 0,
  utilizationAfter: 0.5990323996728818,
  utilizationPrice: 323686.640499199
}
{
  utilizationBefore: 0.461362133874553,
  utilizationAfter: 0.5686047789724876,
  utilizationPrice: 74726.58464112617
}
{
  portfolioDelta: 147.848968608733,
  portfolioGamma: -0.1538578731077324,
  portfolioTheta: 3137.4524202195425,
  portfolioVega: -1649.4313576722554,
  callsPutsValue: 406959.4397028799,
  bsCallsPutsValue: 398413.22514032514
}
NAV after withdraw is: 2016350067173120100000000
      ✔ Sc2. pauses trading and executes epoch (1331ms)
    Sc 3. Single large option purchase and update and another option purchase checks
      ✔ Sc3. Adds liquidity to the liquidityPool (1241ms)
      ✔ Sc3. Another adds liquidity to the liquidityPool (637ms)
      ✔ Sc3. pauses trading and executes epoch (994ms)
{ collateralAllocatedBefore: BigNumber { value: "0" } }
{ quote: '356055.60541235485059424' }
Pool should now have delta value of: 222850258183497507200
Pool should now have an options portfolio value of (or liabilities): 356055.5695312279
NAV after issuance should be: 2000000000000000000000000
NAV after issuance is: 2000000000000000000000000
{ collateralAllocatedAfter: BigNumber { value: "1086994811040" } }
```

```
      ✔ Sc3. LP Writes a WETH/USD put collateralized by USD for premium to the attacker (1363ms)
{
  utilizationBefore: 0,
  utilizationAfter: 0.5434974055,
  utilizationPrice: 323686.8210026461
}
{
  utilizationBefore: 0,
  utilizationAfter: 0.5434974055,
  utilizationPrice: 323686.8210026461
}
{
  portfolioDelta: 222.85028854890703,
  portfolioGamma: -0.12854325965329505,
  portfolioTheta: 2580.4151214238923,
  portfolioVega: -1373.1575705254616,
  callsPutsValue: 356055.5031029107,
  bsCallsPutsValue: 323686.8210026461
}
{
  beforeNAV: BigNumber { value: "200000000000000000000000" },
  afterNAV: BigNumber { value: "200000036125089300000000" }
}
{ collateralAllocated: BigNumber { value: "1086994811040" } }
      ✔ Sc3. should update NAV after fulfill (546ms)
{ collateralAllocatedBefore: BigNumber { value: "1086994811040" } }
{ quote: '67253.97348351272258217' }
Pool should now have delta value of: 297851665304699286900
Pool should now have an options portfolio value of (or liabilities): 423309.5190084159
NAV after issuance should be: 200000000000000000000000
NAV after issuance is: 200000036125089300000000
{ collateralAllocatedAfter: BigNumber { value: "1339664439167" } }
      ✔ Sc3. LP Writes a WETH/USD call collateralized by USD for premium to the attacker (1343ms)
      ✔ Sc3. initiates withdraw liquidity (662ms)
{
  utilizationBefore: 0,
  utilizationAfter: 0.5990324016536077,
  utilizationPrice: 323686.640499199
}
{
  utilizationBefore: 0.461362133874553,
  utilizationAfter: 0.5686047789724876,
  utilizationPrice: 74726.58464112617
}
{
  utilizationBefore: 0,
  utilizationAfter: 0.5990324016536077,
  utilizationPrice: 323686.640499199
}
{
  utilizationBefore: 0.461362133874553,
  utilizationAfter: 0.5686047789724876,
  utilizationPrice: 74726.58464112617
}
{
  portfolioDelta: 147.848968608733,
  portfolioGamma: -0.1538578731077324,
  portfolioTheta: 3137.4524202195425,
  portfolioVega: -1649.4313576722554,
  callsPutsValue: 406959.4397028799,
  bsCallsPutsValue: 398413.22514032514
}
NAV after withdraw is: 2016350061337120100000000
      ✔ Sc2. pauses trading and executes epoch (1160ms)

  Liquidity Pools
    ✔ Succeeds: sets utilization skew params correctly (135ms)
    ✔ Succeeds: User 1: Deposit to the liquidityPool (1227ms)
    ✔ Succeeds: pauses trading (62ms)
    ✔ Succeeds: execute epoch (1037ms)
    ✔ deploys the hedging reactor (184ms)
    ✔ sets reactor address on LP contract (150ms)
    ✔ Returns a quote for a ETH/USD put with utilization (352ms)
    ✔ Returns a quote for a ETH/USD put to buy (309ms)
    ✔ Reverts: Push to price deviation threshold to cause quote to fail (151ms)
    ✔ Reverts: Push to price deviation threshold to cause quote to fail other way (93ms)
    ✔ Reverts: Push to time deviation threshold to cause quote to fail (93ms)
    ✔ reverts when attempting to write ETH/USD puts with expiry outside of limit (542ms)
    ✔ reverts when attempting to write a ETH/USD put with strike outside of limit (637ms)
    ✔ reverts when attempting to write ETH/USD call with expiry outside of limit (483ms)
    ✔ reverts when attempting to write a ETH/USD call with strike outside of limit (627ms)
    ✔ can compute portfolio delta
    ✔ LP Writes a ETH/USD put for premium (1208ms)
    ✔ can issue a put series (90ms)
    ✔ can issue a call series (155ms)
    ✔ can compute portfolio delta (76ms)
    ✔ writes more options for an existing series (857ms)
    ✔ pauses and unpauses handler contract
    ✔ LP writes another ETH/USD put that expires later (1575ms)
    ✔ adds address to the buyback whitelist
    1) LP can buy back option to reduce open interest
    ✔ fails if buyback token address is invalid
    ✔ buys back an option from a non-whitelisted address if it moves delta closer to zero (579ms)
    ✔ can compute portfolio delta (108ms)
    ✔ reverts if option collateral exceeds buffer limit (550ms)
    ✔ reverts when non-admin calls rebalance function (46ms)
    ✔ reverts when rebalance delta too small
    ✔ returns zero when hedging positive delta when reactor has no funds (233ms)
    ✔ Returns a quote for ETH/USD call with utilization (347ms)
    ✔ Creates a buy order (488ms)
    ✔ creates a custom strangle order (515ms)
    ✔ Cant make a buy order if not admin
    ✔ Create buy order reverts if price is zero
    ✔ Create buy order reverts if order expiry too long
    ✔ cant exercise order if not buyer
    ✔ Executes a buy order (796ms)
    ✔ executes a strangle (2226ms)
    ✔ does not buy back an option from a non-whitelisted address if it moves delta away to zero (295ms)
    ✔ Cannot complete buy order after expiry (230ms)
    ✔ fails to execute invalid custom orders (1393ms)
    ✔ Can compute IV from volatility skew coefs
    ✔ Succeeds: User 2: Deposit to the liquidityPool (891ms)
    ✔ Succeeds: pauses trading (106ms)
    ✔ Succeeds: execute epoch (1328ms)
    ✔ Succeed: User 1: redeems all shares (436ms)
    ✔ Succeed: User 1: Initiates Withdraw for half owned balance (337ms)
    ✔ pauses and unpauses LP contract (889ms)
    ✔ settles an expired ITM vault (1725ms)
    ✔ settles an expired OTM vault (1812ms)
    ✔ Reverts: tries to sell an expired option back to the pool
    ✔ Reverts: tries to write an option that doesnt exist in the handler
    ✔ updates option params with setter (144ms)
    ✔ adds and deletes a hedging reactor address (877ms)
    ✔ sets new custom order bounds
    ✔ updates collateralCap variable
    ✔ updates maxDiscount variable
    ✔ updates bufferPercentage variable
    ✔ updates riskFreeRate variable (75ms)
    ✔ sets new utilization skew params (77ms)
    ✔ pauses trading (224ms)
    ✔ handler-only functions in Liquidity pool revert if not called by handler (69ms)
    ✔ returns a volatility skew
    ✔ protocol changes feeds

  Liquidity Pool with alpha tests
    Deposit funds into the liquidityPool
      ✔ SUCCEEDS: User 1: Deposit to the liquidityPool (701ms)
      ✔ SUCCEEDS: pauses trading (72ms)
      ✔ Succeeds: execute epoch (1516ms)
    Create and execute a single buy order
      ✔ SUCCEEDS: Creates a buy order (401ms)
      ✔ REVERTS: Cant make a buy order if not admin (81ms)
      ✔ REVERTS: Cant create buy order if price is zero
      ✔ REVERTS: Cant create buy order if order expiry too long
      ✔ REVERTS: cant exercise order if not buyer
      ✔ REVERTS: Cant execute sell order to buyback order
      ✔ SUCCEEDS: Executes a buy order (665ms)
    Create and execute a strangle
      ✔ SUCCEEDS: creates a custom strangle order (472ms)
      ✔ SETUP: fulfill (71ms)
      ✔ SUCCEEDS: executes a strangle (2022ms)
    Create and execute a single buyback order
      ✔ SETUP: Creates a buy order (256ms)
      ✔ SETUP: Executes a buy order (768ms)
      ✔ SUCCEEDS: Creates a buyback order (223ms)
      ✔ REVERTS: Cant make a buyback order if not admin
      ✔ REVERTS: Cant create buyback order if price is zero
      ✔ REVERTS: Cant create buyback order if order expiry too long
      ✔ REVERTS: cant exercise order if not buyer
      ✔ REVERTS: Cant execute buyback order to sell order
      ✔ SUCCEEDS: Executes a buyback order (669ms)
      ✔ SUCCEEDS: Creates a buyback order on the same option (228ms)
      ✔ REVERTS: Doesnt Execute a buyback order for option with no position (396ms)
    Create a buy order and fail to meet order in time
      ✔ SUCCEEDS: Creates a buy order (289ms)
      ✔ REVERTS: Cant execute after order expires
    Create a buy order and spot moves past deviation threshold
      ✔ SUCCEEDS: Creates a buy order (113ms)
      ✔ REVERTS: Cant execute after spot moves too much up
      ✔ REVERTS: Cant execute after spot moves too much down
    Liquidate a position and update stores, make sure stores update properly
      ✔ SETUP: partially liquidates a vault (110ms)
      ✔ SUCCEEDS: sets stores to correct amount of liquidated vault (40ms)
      ✔ REVERTS: cant account series that isnt stored
    Deposit funds into the liquidityPool and withdraw
      ✔ SUCCEEDS: User 2: Deposit to the liquidityPool (643ms)
      ✔ SUCCEEDS: pauses trading (212ms)
      ✔ Succeeds: execute epoch (953ms)
      ✔ SUCCEEDS: User 1: redeems all shares (647ms)
      ✔ SUCCEEDS: User 1: Initiates Withdraw for half owned balance (335ms)

  Liquidity Pools Deposit Withdraw
    Deposit funds into the liquidityPool
      ✔ Succeeds: User 1: Deposit to the liquidityPool (716ms)
      ✔ Succeeds: User 1: Deposit to the liquidityPool again (1268ms)
      ✔ Succeeds: User 2: Deposit to the liquidityPool (673ms)
      ✔ Reverts: User 1: Tries Zero on all functions (47ms)
      ✔ Reverts: User 1: Attempts to redeem before epoch initiation
      ✔ Reverts: User 1: Attempts to initiate withdraw before epoch initiation (55ms)
      ✔ Reverts: User 1: Attempts to complete withdraw before epoch initiation
      ✔ Reverts: execute epoch before pause
      ✔ Succeeds: pauses trading (66ms)
      ✔ Succeeds: execute epoch (937ms)
    Create and execute a single buy order
      ✔ SUCCEEDS: Creates a buy order (486ms)
      ✔ REVERTS: Cant make a buy order if not admin
      ✔ REVERTS: Cant create buy order if price is zero
      ✔ REVERTS: Cant create buy order if order expiry too long
      ✔ REVERTS: cant exercise order if not buyer
      ✔ SUCCEEDS: Executes a buy order (646ms)
    has another deposit
      ✔ Succeeds: User 3: Deposit to the liquidityPool (644ms)
    Users redeem their shares
      ✔ Reverts: User 3: Attempts to redeem before epoch initiation
      ✔ Reverts: User 3: Attempts to initiate withdraw before epoch initiation
      ✔ Reverts: User 3: Attempts to complete withdraw before epoch initiation
      ✔ Succeed: User 1: redeems all shares (429ms)
      ✔ Revert: User 1: redeems all shares again (64ms)
      ✔ Succeed: User 2: redeems partial shares (593ms)
    user initiates withdraw their funds
      ✔ Succeed: User 1: Initiates Withdraw for half owned balance (342ms)
    Create and execute a strangle
      ✔ SUCCEEDS: creates a custom strangle order (469ms)
      ✔ SETUP: fulfill (70ms)
      ✔ SUCCEEDS: executes a strangle (1159ms)
    executes epoch with new position
      ✔ Succeeds: pauses trading (193ms)
      ✔ Succeeds: execute epoch (953ms)
    more users deposit/withdraw
      ✔ Succeeds: User 3: Deposit to the liquidityPool (977ms)
      ✔ Succeeds: User 1: can complete withdrawal (472ms)
      ✔ Succeed: User 1: Initiates Withdraw for half owned balance (510ms)
      ✔ Succeed: User 2: Initiates Withdraw for owned balance with same redeemable balance (665ms)
      ✔ Succeed: User 2: Initiates Withdraw for owned balance again in same epoch (not using redeemable sh
      ✔ Reverts: User 1: cannot complete withdrawal because of epoch not closed
      ✔ Succeeds: pauses trading (199ms)
      ✔ Succeeds: execute epoch (957ms)

  Liquidity Pools Deposit Withdraw
    ✔ Succeeds: User 1: Deposit to the liquidityPool (1222ms)
    ✔ Succeeds: User 1: Deposit to the liquidityPool again (653ms)
    ✔ Succeeds: User 2: Deposit to the liquidityPool (799ms)
    ✔ Reverts: User 1: Tries Zero on all functions (47ms)
    ✔ Reverts: User 1: Attempts to redeem before epoch initiation
    ✔ Reverts: User 1: Attempts to initiate withdraw before epoch initiation (50ms)
    ✔ Reverts: User 1: Attempts to complete withdraw before epoch initiation
    ✔ Reverts: execute epoch before pause
    ✔ Succeeds: pauses trading (60ms)
    ✔ Succeeds: User 1: issues an option (156ms)
    ✔ Succeeds: execute epoch (989ms)
    ✔ Succeeds: User 3: Deposit to the liquidityPool (644ms)
    ✔ Reverts: User 3: Attempts to redeem before epoch initiation
    ✔ Reverts: User 3: Attempts to initiate withdraw before epoch initiation
    ✔ Reverts: User 3: Attempts to complete withdraw before epoch initiation
    ✔ Succeed: User 1: redeems all shares (612ms)
    ✔ Revert: User 1: redeems all shares again
    ✔ Succeed: User 2: redeems partial shares (685ms)
    ✔ Succeed: User 1: Initiates Withdraw for half owned balance (354ms)
    ✔ Succeeds: User 1: LP Writes a ETH/USD put for premium (1171ms)
    ✔ Succeeds: pauses trading (100ms)
    ✔ Reverts: User 1: cant write option (298ms)
    ✔ Reverts: User 1: cant issue and write option (209ms)
    ✔ Succeeds: execute epoch (967ms)
    ✔ Succeeds: User 3: Deposit to the liquidityPool (750ms)
    ✔ Succeeds: User 1: can complete withdrawal (491ms)
    ✔ Succeed: User 1: Initiates Withdraw for half owned balance (506ms)
    ✔ Succeed: User 2: Initiates Withdraw for owned balance with same redeemable balance (668ms)
    ✔ Succeed: User 2: Initiates Withdraw for owned balance again in same epoch (not using redeemable shar
    ✔ Succeeds: User 1: LP Writes a ETH/USD put for premium (1139ms)
    ✔ Reverts: User 1: cannot complete withdrawal because of epoch not closed
    ✔ Succeeds: pauses trading (67ms)
    ✔ Succeeds: execute epoch with not enough funds to execute withdrawals (1258ms)
    ✔ Reverts: User 1: still cannot complete withdrawal because of withdrawal epoch not closed
    ✔ Succeeds: Reduces collateral cap
    ✔ Reverts: User 1: Deposit to the liquidityPool but hits collat cap (51ms)
    ✔ Succeeds: Raises collateral cap (60ms)
    ✔ Succeeds: pauses trading from keeper (142ms)
    ✔ Succeeds: execute epoch from keeper (1735ms)
    ✔ Reverts: pauses trading from unauthorised
    ✔ Reverts: execute epoch from unauthorised

  Liquidity Pools hedging reactor: perps
    ✔ Deposit to the liquidityPool (620ms)
    ✔ pauses trading and executes epoch (1536ms)
    ✔ #deploys rage (4048ms)
    ✔ #deploys the hedging reactor (225ms)
    ✔ #deploy range order (419ms)
    ✔ can compute portfolio delta
    ✔ LP Writes a ETH/USD put for premium (1440ms)
    ✔ LP writes another ETH/USD put that expires later (1227ms)
    ✔ can compute portfolio delta
    ✔ reverts when non-admin calls rebalance function (41ms)
    ✔ hedges positive delta in perp hedging reactor (432ms)
    ✔ Adds additional liquidity from new account (4355ms)
    ✔ pauses trading and executes epoch (5110ms)
    ✔ initiates withdraw liquidity (1214ms)
    ✔ pauses trading and executes epoch (5212ms)
    ✔ LP can redeem shares (370ms)
    ✔ settles an expired ITM vault (1907ms)
    ✔ settles an expired OTM vault (1638ms)
    ✔ Succeed: Perp hedging reactor unwind (3274ms)

  Liquidity Pools hedging reactor: univ3
    ✔ Deposit to the liquidityPool (1309ms)
    ✔ pauses trading and executes epoch (1518ms)
    ✔ deploys the hedging reactor (198ms)
    ✔ can compute portfolio delta
    ✔ LP Writes a ETH/USD call for premium (1210ms)
    ✔ LP writes another ETH/USD call that expires later (1411ms)
    ✔ can compute portfolio delta
    ✔ reverts when non-admin calls rebalance function (41ms)
    ✔ hedges negative delta in hedging reactor (691ms)
    ✔ Adds additional liquidity from new account (831ms)
    ✔ pauses trading and executes epoch (1305ms)
    ✔ initiates withdraw liquidity (954ms)
    ✔ pauses trading and executes epoch (1252ms)
    ✔ LP can redeem shares (480ms)
    ✔ settles an expired ITM vault (1658ms)
    ✔ settles an expired OTM vault (1632ms)
    ✔ Succeed: Hedging reactor unwind (766ms)

  Options protocol
    ✔ Deploys the Option Registry (198ms)
    ✔ Creates a USDC collataralised call option token series (73ms)
    ✔ Reverts: Tries to close oToken series that doesnt have a vault
    ✔ Returns correct oToken when calling getOrDeployOtoken
    ✔ Returns correct oToken when calling getOToken
    ✔ Returns zero addy if option doesnt exist
    ✔ Creates a ETH collataralised call option token series (63ms)
    ✔ opens call option token with USDC (284ms)
    ✔ opens call option token with ETH (305ms)
    ✔ writer transfers part of balance to new account (53ms)
    ✔ receiver attempts to close and transaction should revert
    ✔ opens call option again with USDC (261ms)
    ✔ opens call option again with ETH (254ms)
    ✔ liquidityPool close and transaction succeeds (227ms)
    ✔ reverts liquidityPool because of non-existent series (79ms)
    ✔ liquidityPool close and transaction succeeds ETH options (228ms)
    ✔ should not allow anyone outside liquidityPool to open (72ms)
    ✔ Fails to settle early
    ✔ Fails to redeem early
    ✔ Fails to settle non-existent option
    ✔ Fails to redeem non-existent option
    ✔ #fastforwards time and sets oracle price (126ms)
    ✔ Fails to create a USDC collataralised call option token series when expired
    ✔ Fails to open a USDC collataralised call option token series when expired
    ✔ Fails to close a USDC collataralised call option token series when expired
    ✔ settles when option expires ITM USD collateral (155ms)
    ✔ reverts when attempt to settle again
    ✔ settles when option expires ITM ETH collateral (157ms)
    ✔ writer redeems when option expires ITM USD collateral (127ms)
    ✔ Fails when writer redeems twice
    ✔ writer redeems when option expires ITM ETH collateral (124ms)
    ✔ creates a USDC collateralised call option token series (59ms)
    ✔ creates a ETH collateralised call option token series (55ms)
    ✔ creates a USDC put option token series (57ms)
    ✔ creates a ETH put option token series (55ms)
    ✔ opens put option token position (1099ms)
    ✔ opens an ERC20 call option (395ms)
    ✔ writer transfers part of erc20 call balance to new account (51ms)
    ✔ writer closes not transfered balance on ERC20 call option (163ms)
    ✔ writer transfers part of put balance to new account (50ms)
    ✔ writer closes not transfered balance on put option token (306ms)
    ✔ settles call when option expires OTM (235ms)
    ✔ writer redeems call when option expires OTM (107ms)
    ✔ settles put when option expires ITM (136ms)
    ✔ writer redeems put when option expires ITM (116ms)
    ✔ sets the health threshold
    ✔ gets the series via issuance hash
    ✔ gets the series via series

  Options protocol Vault Health
    ✔ Deploys the Option Registry (220ms)
    ✔ Creates a liquidity pool (63ms)
    ✔ Creates a USDC collataralised call option token series (74ms)
    ✔ Creates a ETH collataralised call option token series (54ms)
    ✔ opens call option token with USDC (928ms)
    ✔ opens call option token with ETH (331ms)
    ✔ opens call option again with USDC (280ms)
    ✔ opens call option again with ETH (281ms)
    ✔ liquidityPool close and transaction succeeds (305ms)
    ✔ liquidityPool close and transaction succeeds ETH options (194ms)
    ✔ moves the price and changes vault health USD (170ms)
    ✔ liquidityPool close and transaction succeeds (311ms)
    ✔ moves the price and changes vault health ETH (172ms)
    ✔ moves the price and changes vault health USD to negative rebalance stage (226ms)
    ✔ readjusts to negative and checks liquidate (237ms)
    ✔ reverts if unauthorised party tries to adjust collateral (79ms)
    ✔ adjusts collateral to get back to positive (346ms)
    ✔ readjusts to negative and checks liquidate for caller adjust (229ms)
    ✔ adjusts collateral caller to get back to positive (324ms)
    ✔ reverts when trying to adjust a healthy vault (83ms)
    ✔ reverts adjustCollateralCaller when trying to adjust a healthy vault (68ms)
    ✔ moves the price and changes vault health ETH to negative rebalance stage (189ms)
    ✔ moves the price and changes vault health USD to positive rebalance stage (173ms)
    ✔ adjusts overcollateralised position (328ms)
    ✔ moves the price and changes vault health ETH to positive rebalance stage (179ms)
    ✔ settles when option expires ITM USD collateral (164ms)
    ✔ settles when option expires ITM ETH collateral (143ms)
    ✔ writer redeems when option expires ITM USD collateral (121ms)
    ✔ writer redeems when option expires ITM ETH collateral (279ms)
    ✔ creates a USDC put option token series (123ms)
    ✔ creates a ETH put option token series (68ms)
    ✔ opens put option token position (499ms)
    ✔ moves the price and changes vault health USD (157ms)
    ✔ moves the price and changes vault health USD to negative rebalance stage (161ms)
    ✔ moves the price and changes vault health USD to positive rebalance stage (160ms)
    ✔ writer closes not transfered balance on put option token (312ms)
    ✔ settles put when option expires ITM (143ms)
    ✔ writer redeems put when option expires ITM (116ms)
    ✔ Creates a USD collataralised call option token series (247ms)
    ✔ moves the price and changes vault health USD to negative rebalance stage (225ms)
    ✔ vault gets liquidated (215ms)
    ✔ Creates a USD collataralised call option token series (213ms)
    ✔ moves the price and changes vault health USD to negative rebalance stage (228ms)
    ✔ vault gets partially liquidated (132ms)
    ✔ vault liquidated remaining amount (213ms)
    ✔ Creates a USD collataralised call option token series (321ms)
    ✔ moves the price and changes vault health USD to negative rebalance stage (474ms)
    ✔ vault gets liquidated by non-holder (392ms)

  Oracle core logic
{
  utilizationBefore: 0,
  utilizationAfter: 0.06978290000000001,
  utilizationPrice: 474.94676488972163
}
    ✔ Sets state with written options (3318ms)
{
  utilizationBefore: 0,
  utilizationAfter: 0.07424309080144204,
  utilizationPrice: 474.9466897441059
}
{
  utilizationBefore: 0.06867564711986293,
  utilizationAfter: 0.1433163533231281,
  utilizationPrice: 477.96812756015674
}
    ✔ Computes portfolio delta after writing a call with intial put option from the pool (586ms)
{
  utilizationBefore: 0,
  utilizationAfter: 0.07923618642779971,
  utilizationPrice: 474.9466145984777
}
{
  utilizationBefore: 0.06867564711986293,
  utilizationAfter: 0.21795705952639327,
  utilizationPrice: 955.9360982176026
}
    ✔ Computes portfolio delta after writing an additional call from an existing pool (808ms)
{
  utilizationBefore: 0,
  utilizationAfter: 0.07912894348351623,
  utilizationPrice: 474.9464643071833
}
{
  utilizationBefore: 0.06869750236185022,
  utilizationAfter: 0.21653311561761357,
  utilizationPrice: 946.3764313476642
}
    ✔ Computes portfolio delta after partial buyback of option (480ms)
{
  utilizationBefore: 0,
  utilizationAfter: 0.07912894348351623,
  utilizationPrice: 474.9458631415
}
{
  utilizationBefore: 0.0806153203253375,
  utilizationAfter: 0.0806153203253375,
  utilizationPrice: 0
}
    ✔ properly computed portfolio delta after liquidation event (319ms)
{
  utilizationBefore: 0,
  utilizationAfter: 0.07912894348351623,
  utilizationPrice: 0
}
{
  utilizationBefore: 0.0806153203253375,
  utilizationAfter: 0.0806153203253375,
  utilizationPrice: 0
}
    ✔ properly computes calls and puts values with expired OTM options (119ms)
{
  utilizationBefore: 0,
  utilizationAfter: 0.08077560651822101,
  utilizationPrice: 0
}
{
  utilizationBefore: 0.08232509534591148,
  utilizationAfter: 0.08232509534591148,
  utilizationPrice: 0
}
{
  utilizationBefore: 1.0000000005515222,
  utilizationAfter: 1.0000000005217013,
  utilizationPrice: 2228.790767353186
}
{
  utilizationBefore: 0,
  utilizationAfter: 0.07679479463418397,
  utilizationPrice: 0
}
{
  utilizationBefore: 0.07819399619519508,
  utilizationAfter: 0.07819399619519508,
  utilizationPrice: 0
}
    ✔ properly computes portfolio value with expired ITM options (2486ms)

  PerpHedgingReactor
    ✔ #deploys dummy LP (55ms)
    ✔ #funds accounts
    ✔ #deploy price feed (150ms)
    ✔ #deploys rage (2930ms)
    ✔ #deploys the hedging reactor (69ms)
    ✔ #deploy range order (1097ms)
    ✔ sets reactor address on LP contract
    ✔ returns 0 if getPoolDenominatedValue if not initialised
    ✔ reverts hedgeDelta if not initialised
    ✔ reverts update if not initialised
    ✔ initialises the reactor (53ms)
[
  BigNumber { value: "1" },
  BigNumber { value: "0" },
  marketValue: BigNumber { value: "1" },
  requiredMargin: BigNumber { value: "0" }
]
BigNumber { value: "0" }
[
  BigNumber { value: "24494772735" },
  BigNumber { value: "7999361222" },
  marketValue: BigNumber { value: "24494772735" },
  requiredMargin: BigNumber { value: "7999361222" }
]
BigNumber { value: "-505227265" }
    ✔ hedges a positive delta when position is zero (855ms)
[
  BigNumber { value: "24494772735" },
  BigNumber { value: "7999361222" },
  marketValue: BigNumber { value: "24494772735" },
  requiredMargin: BigNumber { value: "7999361222" }
]
BigNumber { value: "-505227265" }
[
  BigNumber { value: "1" },
  BigNumber { value: "0" },
  marketValue: BigNumber { value: "1" },
  requiredMargin: BigNumber { value: "0" }
]
BigNumber { value: "0" }
    ✔ hedges delta back to 0 (589ms)
[
  BigNumber { value: "1" },
  BigNumber { value: "0" },
  marketValue: BigNumber { value: "1" },
  requiredMargin: BigNumber { value: "0" }
]
BigNumber { value: "0" }
[
  BigNumber { value: "24494772735" },
  BigNumber { value: "7999361222" },
  marketValue: BigNumber { value: "24494772735" },
  requiredMargin: BigNumber { value: "7999361222" }
]
BigNumber { value: "-505227265" }
    ✔ hedges a positive delta when position is zero again (810ms)
[
  BigNumber { value: "24494772735" },
  BigNumber { value: "7999361222" },
  marketValue: BigNumber { value: "24494772735" },
  requiredMargin: BigNumber { value: "7999361222" }
]
BigNumber { value: "-505227265" }
    ✔ syncs profits (832ms)
    ✔ SUCCEEDS: checkvault health if price goes up (526ms)
    ✔ SUCCEEDS: syncAndUpdate to get vault back on  (890ms)
    ✔ SUCCEEDS: checkvault health if price goes down (560ms)
    ✔ SUCCEEDS: syncAndUpdate to get vault back onto normal (1160ms)
[
  BigNumber { value: "24999802220" },
  BigNumber { value: "7812795756" },
  marketValue: BigNumber { value: "24999802220" },
  requiredMargin: BigNumber { value: "7812795756" }
]
BigNumber { value: "-197780" }
[
  BigNumber { value: "24373346601" },
  BigNumber { value: "7617475862" },
  marketValue: BigNumber { value: "24373346601" },
  requiredMargin: BigNumber { value: "7617475862" }
]
BigNumber { value: "-1653399" }
    ✔ hedges a negative delta (1289ms)
    ✔ getDelta returns correct value
    ✔ gets the portfolio value (339ms)
[
  BigNumber { value: "24373346601" },
  BigNumber { value: "7617475862" },
  marketValue: BigNumber { value: "24373346601" },
  requiredMargin: BigNumber { value: "7617475862" }
]
BigNumber { value: "-1653399" }
[
  BigNumber { value: "24724654573" },
  BigNumber { value: "7739309760" },
  marketValue: BigNumber { value: "24724654573" },
  requiredMargin: BigNumber { value: "7739309760" }
]
BigNumber { value: "-25345427" }
    ✔ hedges a positive delta with sufficient funds (1890ms)
    ✔ hedges a positive delta with insufficient funds (50ms)
[
  BigNumber { value: "24736065566" },
  BigNumber { value: "7736988431" },
  marketValue: BigNumber { value: "24736065566" },
  requiredMargin: BigNumber { value: "7736988431" }
]
    ✔ liquidates usdc held position (366ms)
[
  BigNumber { value: "24735967740" },
  BigNumber { value: "7736988431" },
  marketValue: BigNumber { value: "24735967740" },
  requiredMargin: BigNumber { value: "7736988431" }
]
BigNumber { value: "-14032260" }
    ✔ syncs profits (1321ms)
[
  BigNumber { value: "24750000000" },
  BigNumber { value: "7736988431" },
  marketValue: BigNumber { value: "24750000000" },
  requiredMargin: BigNumber { value: "7736988431" }
]
    ✔ liquidates a bit of position and withdraws sufficient funds (251ms)
    ✔ update fixes balances one way (131ms)
    ✔ update fixes balances other way (388ms)
    ✔ update returns 0 (454ms)
    ✔ update reverts when not called by keeper
    ✔ liquidates all positions and withdraws (324ms)
    ✔ updates healthFactor
    ✔ update health factor reverts if not owner
    ✔ withdraw reverts if not called form liquidity pool
    ✔ hedgeDelta reverts if not called from liquidity pool

  PerpHedgingReactor Sc1
    ✔ #deploys dummy LP (69ms)
    ✔ #funds accounts (66ms)
    ✔ #deploy price feed (260ms)
    ✔ #deploys rage (2970ms)
    ✔ #deploys the hedging reactor (97ms)
    ✔ #deploy range order (517ms)
    ✔ sets reactor address on LP contract
    ✔ initialises the reactor (62ms)
[
  BigNumber { value: "1" },
  BigNumber { value: "0" },
  marketValue: BigNumber { value: "1" },
  requiredMargin: BigNumber { value: "0" }
]
BigNumber { value: "0" }
[
  BigNumber { value: "24476179118" },
  BigNumber { value: "7999361222" },
  marketValue: BigNumber { value: "24476179118" },
  requiredMargin: BigNumber { value: "7999361222" }
]
BigNumber { value: "-523820882" }
    ✔ hedges a negative delta when position is zero (1002ms)
    ✔ SUCCEEDS: checkvault health if price goes up (639ms)
    ✔ SUCCEEDS: syncAndUpdate to get vault back on  (1604ms)
    ✔ SUCCEEDS: checkvault health if price goes down (566ms)
    ✔ SUCCEEDS: syncAndUpdate to get vault back onto normal (1164ms)
[
  BigNumber { value: "25000171363" },
  BigNumber { value: "8192839133" },
  marketValue: BigNumber { value: "25000171363" },
  requiredMargin: BigNumber { value: "8192839133" }
]
BigNumber { value: "171363" }
[
  BigNumber { value: "25623878614" },
  BigNumber { value: "8397660112" },
  marketValue: BigNumber { value: "25623878614" },
  requiredMargin: BigNumber { value: "8397660112" }
]
BigNumber { value: "-1121386" }
    ✔ hedges more negative delta (1118ms)
[
  BigNumber { value: "25623878614" },
  BigNumber { value: "8397660112" },
  marketValue: BigNumber { value: "25623878614" },
  requiredMargin: BigNumber { value: "8397660112" }
]
BigNumber { value: "-1121386" }
    ✔ syncs profits (790ms)
[
  BigNumber { value: "25625000000" },
  BigNumber { value: "8402699968" },
  marketValue: BigNumber { value: "25625000000" },
  requiredMargin: BigNumber { value: "8402699968" }
]
BigNumber { value: "0" }
[
  BigNumber { value: "24998802532" },
  BigNumber { value: "8197756066" },
  marketValue: BigNumber { value: "24998802532" },
  requiredMargin: BigNumber { value: "8197756066" }
]
BigNumber { value: "-1197468" }
    ✔ hedges a positive delta (1205ms)
    ✔ getDelta returns correct value
    ✔ gets the portfolio value (334ms)
[
  BigNumber { value: "24998802532" },
  BigNumber { value: "8197756066" },
  marketValue: BigNumber { value: "24998802532" },
  requiredMargin: BigNumber { value: "8197756066" }
]
BigNumber { value: "-1197468" }
[
  BigNumber { value: "25348636044" },
  BigNumber { value: "8315731720" },
  marketValue: BigNumber { value: "25348636044" },
  requiredMargin: BigNumber { value: "8315731720" }
```

```
]
BigNumber { value: "-26363956" }
    ✔ hedges a negative delta with sufficient funds (1032ms)
    ✔ hedges a negative delta with insufficient funds (59ms)
[
  BigNumber { value: "25365356845" },
  BigNumber { value: "8319058512" },
  marketValue: BigNumber { value: "25365356845" },
  requiredMargin: BigNumber { value: "8319058512" }
]
    ✔ liquidates a bit of position and withdraws sufficient funds (299ms)
    ✔ update fixes balances one way (136ms)
    ✔ update fixes balances other way (328ms)
    ✔ update returns 0 (77ms)
    ✔ liquidates all positions and withdraws (317ms)

  PerpHedgingReactor Sc2
    ✔ #deploys dummy LP (82ms)
    ✔ #funds accounts (40ms)
    ✔ #deploy price feed (141ms)
    ✔ #deploys rage (2880ms)
    ✔ #deploys the hedging reactor (74ms)
    ✔ #deploy range order (525ms)
    ✔ sets reactor address on LP contract
    ✔ initialises the reactor (63ms)
[
  BigNumber { value: "1" },
  BigNumber { value: "0" },
  marketValue: BigNumber { value: "1" },
  requiredMargin: BigNumber { value: "0" }
]
BigNumber { value: "0" }
[
  BigNumber { value: "24476179118" },
  BigNumber { value: "7999361222" },
  marketValue: BigNumber { value: "24476179118" },
  requiredMargin: BigNumber { value: "7999361222" }
]
BigNumber { value: "-523820882" }
    ✔ hedges a negative delta when position is zero (2705ms)
[
  BigNumber { value: "25443950575" },
  BigNumber { value: "8192839133" },
  marketValue: BigNumber { value: "25443950575" },
  requiredMargin: BigNumber { value: "8192839133" }
]
BigNumber { value: "443950575" }
[
  BigNumber { value: "29142699445" },
  BigNumber { value: "8397660112" },
  marketValue: BigNumber { value: "29142699445" },
  requiredMargin: BigNumber { value: "8397660112" }
]
BigNumber { value: "442699445" }
    ✔ hedges more negative delta (2149ms)
[
  BigNumber { value: "29168326587" },
  BigNumber { value: "8402699968" },
  marketValue: BigNumber { value: "29168326587" },
  requiredMargin: BigNumber { value: "8402699968" }
]
BigNumber { value: "468326587" }
[
  BigNumber { value: "29467184102" },
  BigNumber { value: "8197756066" },
  marketValue: BigNumber { value: "29467184102" },
  requiredMargin: BigNumber { value: "8197756066" }
]
BigNumber { value: "467184102" }
    ✔ hedges a positive delta (1115ms)
[
  BigNumber { value: "29467184102" },
  BigNumber { value: "8197756066" },
  marketValue: BigNumber { value: "29467184102" },
  requiredMargin: BigNumber { value: "8197756066" }
]
BigNumber { value: "467184102" }
    ✔ syncs profits (771ms)
    ✔ getDelta returns correct value
    ✔ gets the portfolio value (332ms)
[
  BigNumber { value: "29000312006" },
  BigNumber { value: "8192839133" },
  marketValue: BigNumber { value: "29000312006" },
  requiredMargin: BigNumber { value: "8192839133" }
]
BigNumber { value: "312006" }
[
  BigNumber { value: "30449709240" },
  BigNumber { value: "8315731720" },
  marketValue: BigNumber { value: "30449709240" },
  requiredMargin: BigNumber { value: "8315731720" }
]
BigNumber { value: "-290760" }
    ✔ hedges a negative delta with sufficient funds (1367ms)
    ✔ hedges a negative delta with insufficient funds (56ms)
[
  BigNumber { value: "30466495805" },
  BigNumber { value: "8319058512" },
  marketValue: BigNumber { value: "30466495805" },
  requiredMargin: BigNumber { value: "8319058512" }
]
```

```
BigNumber { value: "16495805" }
    ✔ syncs profits (804ms)
[
  BigNumber { value: "30450347643" },
  BigNumber { value: "8319058512" },
  marketValue: BigNumber { value: "30450347643" },
  requiredMargin: BigNumber { value: "8319058512" }
]
    ✔ liquidates a bit of position and withdraws sufficient funds (301ms)
[
  BigNumber { value: "30450408925" },
  BigNumber { value: "8319058512" },
  marketValue: BigNumber { value: "30450408925" },
  requiredMargin: BigNumber { value: "8319058512" }
]
BigNumber { value: "408925" }
    ✔ update fixes balances one way (683ms)
    ✔ update fixes balances other way (323ms)
    ✔ update returns 0 (84ms)
    ✔ liquidates all positions and withdraws (220ms)

  APVF gas tests
    ✔ SETUP: make all settings lenient (138ms)
    Spin up a bunch of options and try a fulfill
      ✔ SETUP: Spin up a bunch of options (12791ms)
      ✔ SUCCEEDS: Calls fulfill on the options (2109ms)
    Try a migration with all the options
      ✔ SETUP: Make a new portfolio values feed (156ms)
      ✔ SUCCEEDS: Tries to migrate to a new portfolio values feed (538ms)
      ✔ SUCCEEDS: Checks the new fulfill are the same as the old fulfill (4390ms)
      ✔ SETUP: reconfigure original portfolio values feed (45ms)
    Expire some of the options and try a clean
      ✔ SETUP: fastforward 3 days so options have expired (69ms)
      ✔ SUCCEEDS: Cleans one expired option manually (123ms)
      ✔ FAILS: Cleans one expired option manually with incorrect address (47ms)
      ✔ FAILS: Cleans one option that is not expired (59ms)
      ✔ SUCCEEDS: Cleans all expired options (178ms)
    Expire some of the options at the end and try a clean
      ✔ SETUP: writes some options at the end of the array that expire soon (1899ms)
      ✔ SETUP: increments option series already stored (1168ms)
      ✔ SETUP: fastforward 3 days so options have expired
      ✔ SUCCEEDS: Cleans all expired options (127ms)
    Expire some of the options and try a fulfill without first cleaning
      ✔ SETUP: fastforward 3 days so options have expired
      ✔ FAILS: Fulfill fails because of expired options not cleaned (462ms)
      ✔ SUCCEEDS: Cleans all expired options (71ms)
      ✔ SUCCEEDS: Fulfills correctly (2940ms)
    Reduce the short exposure on a series and check fulfill
      ✔ SUCCEEDS: reduces the short exposure on a series and checks the fulfill (3329ms)
    Add long exposure and check fulfill
      ✔ SUCCEEDS: increases the long exposure on a series and checks the fulfill (3537ms)
      ✔ SETUP: removes all short from index 10 (3203ms)
      ✔ REVERTS: cant account liquidated series with no short
      ✔ REVERTS: cant account with no vault
    Access Control checks
      ✔ SUCCEEDS: set liquidity pool
      ✔ FAILS: set liquidity pool when not approved
      ✔ SUCCEEDS: set protocol
      ✔ FAILS: set protocol when not approved
      ✔ SUCCEEDS: set rfr
      ✔ FAILS: set rfr when not approved
      ✔ SUCCEEDS: set keeper
      ✔ SUCCEEDS: remove keeper
      ✔ FAILS: set keeper when not approved
      ✔ SUCCEEDS: set handler
      ✔ SUCCEEDS: remove handler
      ✔ FAILS: set keeper when not approved
      ✔ FAILS: update stores if not handler
      ✔ FAILS: sync looper if not handler
      ✔ FAILS: clean looper manually if not handler
      ✔ FAILS: migration if not governance

  Price Feed
    ✔ Should deploy price feed (214ms)
    ✔ Should return a price quote
    ✔ Should return a normalized price quote (42ms)
    ✔ Should return a normalised price quote on e18 decimals (57ms)
    ✔ Should revert for a non-existent price quote
    ✔ Should revert for a non-existent normalised price quote

  UniswapV3HedgingReactor
    ✔ deploys the dummy LP contract (69ms)
    ✔ funds the LP contract with a million USDC
    ✔ Should deploy price feed (175ms)
    ✔ deploys the hedging reactor (94ms)
    ✔ updates minAmount parameter
    ✔ sets reactor address on LP contract
    ✔ changes nothing if no ETH balance and hedging positive delta (54ms)
    ✔ hedges a negative delta (94ms)
    ✔ getDelta returns correct value
    ✔ gets the portfolio value
    ✔ hedges a positive delta with sufficient funds (80ms)
    ✔ hedges a positive delta with insufficient funds (70ms)
    ✔ withdraws funds without liquidation (97ms)
    ✔ liquidates WETH and withdraws sufficient funds (617ms)
    ✔ liquidates all ETH and withdraws but does not have enough funds
    ✔ update changes no balances
    ✔ updates poolFee
    ✔ update pool fee reverts if not owner
    ✔ withdraw reverts if not called form liquidity pool
    ✔ hedgeDelta reverts if not called from liquidity pool
  537 passing (7m)
  1 failing

  1) Liquidity Pools
       LP can buy back option to reduce open interest:
     AssertionError: expected -0.00100000000009004 to be within -0.001..0.001
      at /Users/andreisimion/playground/cryptocurrency/akiratech/review-rysk-dynamic-hedging-2022-08/code/
      at step (test/LiquidityPool.ts:52:23)
      at Object.next (test/LiquidityPool.ts:33:53)
      at fulfilled (test/LiquidityPool.ts:24:58)
      at runMicrotasks (<anonymous>)
      at processTicksAndRejections (node:internal/process/task_queues:96:5)
      at runNextTicks (node:internal/process/task_queues:65:3)
      at listOnTimeout (node:internal/timers:528:9)
      at processTimers (node:internal/timers:502:7)



------------------------------------------------------------------|----------|----------|----------|------
File                                                              |  % Stmts | % Branch |  % Funcs |  % Li
------------------------------------------------------------------|----------|----------|----------|------
 contracts/                                                       |       94 |     86.4 |    92.61 |    92
  Accounting.sol                                                  |      100 |    86.67 |      100 |    95
  AlphaOptionHandler.sol                                          |      100 |       90 |      100 |    97
  AlphaPortfolioValuesFeed.sol                                    |      100 |      100 |      100 |
  Authority.sol                                                   |      100 |      100 |      100 |
  LiquidityPool.sol                                               |    98.31 |    80.88 |      100 |    95
  OptionHandler.sol                                               |    97.75 |    91.18 |    94.12 |    97
  OptionRegistry.sol                                              |      100 |    87.04 |      100 |
  PortfolioValuesFeed.sol                                         |        0 |        0 |        0 |
  PriceFeed.sol                                                   |      100 |      100 |      100 |
  Protocol.sol                                                    |      100 |      100 |      100 |
  VolatilityFeed.sol                                              |       90 |    83.33 |    85.71 |    85
 contracts/hedging/                                               |    91.11 |    72.37 |     96.3 |    88
  PerpHedgingReactor.sol                                          |    90.98 |    72.41 |    93.33 |     8
  UniswapV3HedgingReactor.sol                                     |    91.38 |    72.22 |      100 |    89
 contracts/interfaces/                                            |      100 |      100 |      100 |
  AddressBookInterface.sol                                        |      100 |      100 |      100 |
  AggregatorV3Interface.sol                                       |      100 |      100 |      100 |
  GammaInterface.sol                                              |      100 |      100 |      100 |
  IAccounting.sol                                                 |      100 |      100 |      100 |
  IAuthority.sol                                                  |      100 |      100 |      100 |
  IHedgingReactor.sol                                             |      100 |      100 |      100 |
  ILiquidityPool.sol                                              |      100 |      100 |      100 |
  IMarginCalculator.sol                                           |      100 |      100 |      100 |
  IOptionRegistry.sol                                             |      100 |      100 |      100 |
  IOracle.sol                                                     |      100 |      100 |      100 |
  IPortfolioValuesFeed.sol                                        |      100 |      100 |      100 |
  I_ERC20.sol                                                     |      100 |      100 |      100 |
  WETH.sol                                                        |      100 |      100 |      100 |
 contracts/libraries/                                             |    95.22 |    79.63 |       96 |    94
  AccessControl.sol                                               |     62.5 |    83.33 |       80 |    66
  BlackScholes.sol                                                |      100 |      100 |      100 |
  CustomErrors.sol                                                |      100 |      100 |      100 |
  EnumerableSet.sol                                               |    92.86 |       50 |      100 |    93
  NormalDist.sol                                                  |      100 |      100 |      100 |
  OptionsCompute.sol                                              |    91.18 |    77.27 |      100 |    86
  OpynInteractions.sol                                            |      100 |      100 |      100 |
  SafeTransferLib.sol                                             |    77.78 |       75 |       80 |    78
  Types.sol                                                       |      100 |      100 |      100 |
 contracts/mocks/                                                 |    86.36 |       50 |    88.89 |    82
  MockPortfolioValuesFeed.sol                                     |    86.36 |       50 |    88.89 |    82
 contracts/packages/opyn/                                         |        0 |        0 |        0 |
  Migrations.sol                                                  |        0 |        0 |        0 |
 contracts/packages/opyn/core/                                    |        0 |        0 |        0 |
  AddressBook.sol                                                 |        0 |        0 |        0 |
  Controller.sol                                                  |        0 |        0 |        0 |
  MarginCalculator.sol                                            |        0 |        0 |        0 |
  MarginPool.sol                                                  |        0 |        0 |        0 |
  Oracle.sol                                                      |        0 |        0 |        0 |
  Otoken.sol                                                      |        0 |        0 |        0 |
  OtokenFactory.sol                                               |        0 |        0 |        0 |
  OtokenSpawner.sol                                               |        0 |      100 |        0 |
  Whitelist.sol                                                   |        0 |        0 |        0 |
 contracts/packages/opyn/external/callees/                        |        0 |      100 |        0 |
  PermitCallee.sol                                                |        0 |      100 |        0 |
 contracts/packages/opyn/external/canonical-weth/                 |        0 |        0 |        0 |
  WETH9.sol                                                       |        0 |        0 |        0 |
 contracts/packages/opyn/external/proxies/                        |        0 |        0 |        0 |
  PayableProxyController.sol                                      |        0 |        0 |        0 |
 contracts/packages/opyn/interfaces/                              |      100 |      100 |      100 |
  AddressBookInterface.sol                                        |      100 |      100 |      100 |
  AggregatorInterface.sol                                         |      100 |      100 |      100 |
  CTokenInterface.sol                                             |      100 |      100 |      100 |
  CalleeInterface.sol                                             |      100 |      100 |      100 |
  ERC20Interface.sol                                              |      100 |      100 |      100 |
  MarginCalculatorInterface.sol                                   |      100 |      100 |      100 |
  MarginPoolInterface.sol                                         |      100 |      100 |      100 |
  OpynPricerInterface.sol                                         |      100 |      100 |      100 |
  OracleInterface.sol                                             |      100 |      100 |      100 |
  OtokenInterface.sol                                             |      100 |      100 |      100 |
  WETH9Interface.sol                                              |      100 |      100 |      100 |
  WSTETHInterface.sol                                             |      100 |      100 |      100 |
  WhitelistInterface.sol                                          |      100 |      100 |      100 |
  YearnVaultInterface.sol                                         |      100 |      100 |      100 |
  ZeroXExchangeInterface.sol                                      |      100 |      100 |      100 |
 contracts/packages/opyn/libs/                                    |    77.97 |    48.11 |    80.65 |    77
  Actions.sol                                                     |    91.67 |    45.83 |    88.89 |    91
  FixedPointInt256.sol                                            |    77.14 |       75 |    78.57 |
  MarginVault.sol                                                 |    66.67 |    42.86 |    66.67 |    66
  SignedConverter.sol                                             |       80 |       50 |      100 |
 contracts/packages/opyn/mocks/                                    |     2.48 |        0 |     4.21 |     2
  Mock0xERC20Proxy.sol                                             |        0 |      100 |        0 |
  Mock0xExchange.sol                                               |        0 |      100 |        0 |
  MockAddressBook.sol                                              |        0 |      100 |        0 |
  MockCToken.sol                                                   |        0 |      100 |        0 |
  MockCUSDC.sol                                                    |        0 |      100 |        0 |
  MockChainlinkAggregator.sol                                      |    66.67 |        0 |       80 |    66
  MockController.sol                                               |        0 |      100 |        0 |
  MockDumbERC20.sol                                                |        0 |        0 |        0 |
  MockERC20.sol                                                    |        0 |      100 |        0 |
  MockOracle.sol                                                   |        0 |        0 |        0 |
  MockOtoken.sol                                                   |        0 |      100 |        0 |
  MockPermitERC20.sol                                              |        0 |      100 |        0 |
  MockPricer.sol                                                   |        0 |      100 |        0 |
  MockWSTETHToken.sol                                              |        0 |      100 |        0 |
  MockWhitelistModule.sol                                          |        0 |        0 |        0 |
  MockYToken.sol                                                   |        0 |      100 |        0 |
 contracts/packages/opyn/new/                                      |    73.46 |    50.38 |    62.16 |    73
  NewCalculator.sol                                                |    76.36 |    57.02 |    64.86 |    76
  NewController.sol                                                |    71.62 |    45.21 |    58.93 |    70
  NewMarginCalculatorInterface.sol                                 |      100 |      100 |      100 |
  NewWhitelist.sol                                                 |    67.57 |       50 |    66.67 |    68
 contracts/packages/opyn/packages/                                 |        0 |        0 |        0 |
  BokkyPooBahsDateTimeLibrary.sol                                  |        0 |      100 |        0 |
  Spawn.sol                                                        |        0 |        0 |        0 |
 contracts/packages/opyn/packages/oz/                              |    36.36 |    24.14 |    36.84 |    34
  Address.sol                                                      |        0 |        0 |        0 |
  Context.sol                                                      |       50 |      100 |       50 |    33
  Create2.sol                                                      |        0 |        0 |        0 |
  IERC20.sol                                                       |      100 |      100 |      100 |
  Ownable.sol                                                      |       40 |       25 |       40 |    45
  ReentrancyGuard.sol                                              |        0 |        0 |        0 |
  SafeERC20.sol                                                    |        0 |        0 |        0 |
  SafeMath.sol                                                     |      100 |    58.33 |      100 |
  SignedSafeMath.sol                                               |       75 |    42.86 |       75 |
  Strings.sol                                                      |        0 |        0 |        0 |
 contracts/packages/opyn/packages/oz/upgradeability/               |    21.59 |    20.59 |    17.07 |    23
  ERC20Upgradeable.sol                                             |        0 |        0 |        0 |
  IERC20Upgradeable.sol                                            |      100 |      100 |      100 |
  Initializable.sol                                                |      100 |    83.33 |      100 |
  OwnableUpgradeSafe.sol                                           |    45.45 |       25 |       50 |
  OwnedUpgradeabilityProxy.sol                                     |        0 |        0 |        0 |
  Proxy.sol                                                        |        0 |        0 |        0 |
  ReentrancyGuardUpgradeSafe.sol                                   |       80 |       50 |    66.67 |    83
  UpgradeabilityProxy.sol                                          |        0 |        0 |        0 |
 contracts/packages/opyn/packages/oz/upgradeability/GSN/           |    33.33 |      100 |       50 |
  ContextUpgradeable.sol                                           |    33.33 |      100 |       50 |
 contracts/packages/opyn/packages/oz/upgradeability/cryptography/ |        0 |        0 |        0 |
  ECDSAUpgradeable.sol                                             |        0 |        0 |        0 |
 contracts/packages/opyn/packages/oz/upgradeability/erc20-permit/ |        0 |        0 |        0 |
  EIP712Upgradeable.sol                                            |        0 |      100 |        0 |
  ERC20PermitUpgradeable.sol                                      |        0 |        0 |        0 |
  IERC20PermitUpgradeable.sol                                     |      100 |      100 |      100 |
 contracts/packages/opyn/packages/oz/upgradeability/math/         |        0 |        0 |        0 |
  SafeMathUpgradeable.sol                                         |        0 |        0 |        0 |
 contracts/packages/opyn/packages/oz/upgradeability/utils/        |        0 |      100 |        0 |
  CountersUpgradeable.sol                                         |        0 |      100 |        0 |
 contracts/packages/opyn/pricers/                                 |    18.92 |    13.04 |       15 |    17
  ChainlinkPricer.sol                                             |    58.33 |     37.5 |       50 |    54
  CompoundPricer.sol                                              |        0 |        0 |        0 |
  WstethPricer.sol                                                |        0 |        0 |        0 |
  YearnPricer.sol                                                 |        0 |        0 |        0 |
 contracts/packages/opyn/tests/                                   |     1.28 |        0 |     1.79 |     1
  ActionTester.sol                                                |        0 |      100 |        0 |
  CalculatorTester.sol                                            |        0 |      100 |        0 |
  CallTester.sol                                                  |        0 |      100 |        0 |
  CalleeAllowanceTester.sol                                       |        0 |      100 |        0 |
  FixedPointInt256Tester.sol                                      |        0 |      100 |        0 |
  FlashUnwrap.sol                                                 |        0 |        0 |        0 |
  ForceSend.sol                                                   |      100 |      100 |      100 |
  MarginVaultTester.sol                                           |        0 |      100 |        0 |
  OtokenImplV1.sol                                                |        0 |      100 |        0 |
  SignedConverterTester.sol                                       |        0 |      100 |        0 |
  UpgradeableContractV1.sol                                       |        0 |      100 |        0 |
  UpgradeableContractV2.sol                                       |        0 |      100 |        0 |
 contracts/tokens/                                                |    27.14 |     6.25 |       24 |    28
  ERC20.sol                                                       |       76 |       25 |    66.67 |    75
  MintableERC20.sol                                               |        0 |        0 |        0 |
  WETH.sol                                                        |        0 |        0 |        0 |
 contracts/utils/                                                 |    88.89 |       75 |    82.35 |    89
  BlackScholesTest.sol                                            |      100 |      100 |      100 |
  LiquidityPoolAdjustCollateralTest.sol                           |    76.92 |      100 |     62.5 |    76
  OracleMock.sol                                                  |       80 |      100 |       80 |
  PerpHedgingTest.sol                                             |    88.89 |      100 |     87.5 |    88
  RealTokenMock.sol                                               |      100 |      100 |        0 |
  ReentrancyGuard.sol                                             |      100 |       50 |      100 |
  UniswapV3HedgingTest.sol                                        |      100 |      100 |      100 |
  Volatility.sol                                                  |      100 |      100 |      100 |
------------------------------------------------------------------|----------|----------|----------|------
All files                                                         |    52.74 |    39.47 |    43.58 |    53
------------------------------------------------------------------|----------|----------|----------|------

> Istanbul reports written to ./coverage/ and ./coverage.json
Error in plugin solidity-coverage: ❌ 1 test(s) failed under coverage.

For more info run Hardhat with --show-stack-traces
```

NB: I applied a patch from the Rysk team to the `package.json` file, that fixed some of the reverts in the tests: https://github.com/rysk-finance/dynamic-hedging/commit/239a964f3494b57e40d23bc04f20ee92057f8a15

License

This report falls under the terms described in the included LICENSE.