```
1 /**
                                                             1 /**
 2 *Submitted for verification at Etherscan.io on 202
                                                             2 *Submitted for verification at Etherscan.io on 202
   1-06-12
                                                               1-06-12
 3 */
 5 // SPDX-License-Identifier: MIT
                                                             5 // SPDX-License-Identifier: MIT
 6 pragma solidity 0.7.5;
                                                             6 pragma solidity 0.7.5;
8 /**
                                                             8 /**
   * @dev Interface of the ERC20 standard as defined
                                                                * Odey Interface of the FRC20 standard as defined
                                                             9
    in the EIP.
                                                                 in the EIP.
10 */
                                                            10 */
11 interface IERC20 {
                                                            11 interface IERC20 {
        * @dev Returns the amount of tokens in existen
                                                                    * @dev Returns the amount of tokens in existen
13
                                                            13
   ce.
                                                                ce.
14
                                                            14
                                                                   function totalSupply() external view returns (u
      function totalSupply() external view returns (u
15
                                                            15
   int256);
                                                                int256);
16
                                                            16
17
                                                            17
       * @dev Returns the amount of tokens owned by `
                                                                    * @dev Returns the amount of tokens owned by `
18
                                                            18
   account`.
                                                                account`.
19
                                                            19
       function balanceOf(address account) external vi
                                                                    function balanceOf(address account) external vi
20
                                                            20
   ew returns (uint256);
                                                                ew returns (uint256);
21
22
       * @dev Moves `amount` tokens from the caller's
                                                                    * @dev Moves `amount` tokens from the caller's
23
                                                            23
   account to `recipient`.
                                                                account to `recipient`.
                                                            24
24
        ^{\star} Returns a boolean value indicating whether t
                                                                    * Returns a boolean value indicating whether t
25
                                                            25
   he operation succeeded.
                                                                he operation succeeded.
26
                                                            26
27
        * Emits a {Transfer} event.
                                                                    * Emits a {Transfer} event.
28
                                                            28
29
       function transfer(address recipient, uint256 am
                                                            29
                                                                   function transfer(address recipient, uint256 am
   ount) external returns (bool);
                                                                ount) external returns (bool);
31
                                                            31
       * @dev Returns the remaining number of tokens
                                                                    * @dev Returns the remaining number of tokens
32
                                                            32
                                                                 that `spender` will be
    that `spender` will be
       * allowed to spend on behalf of `owner` throug
                                                                    * allowed to spend on behalf of `owner` throug
   h {transferFrom}. This is
                                                                h {transferFrom}. This is
       * zero by default.
                                                                    * zero bv default.
34
                                                            34
35
                                                            35
       * This value changes when {approve} or {transf
                                                                    * This value changes when {approve} or {transf
36
                                                            36
   erFrom} are called.
                                                                erFrom} are called.
       function allowance(address owner, address spend
                                                                   function allowance(address owner, address spend
   er) external view returns (uint256);
                                                                er) external view returns (uint256);
39
                                                            39
40
                                                            40
       * @dev Sets `amount` as the allowance of `spen
                                                                    * @dev Sets `amount` as the allowance of `spen
41
                                                            41
                                                                der` over the caller's tokens.
   der` over the caller's tokens.
42
                                                             42
        * Returns a boolean value indicating whether t
                                                                    * Returns a boolean value indicating whether t
   he operation succeeded.
                                                                he operation succeeded.
        ^{\star} IMPORTANT: Beware that changing an allowance \,
                                                                    * IMPORTANT: Beware that changing an allowance
```

with this method brings the risk

w allowance by unfortunate

* that someone may use both the old and the ne

with this method brings the risk

w allowance by unfortunate

* that someone may use both the old and the ne

```
* transaction ordering. One possible solution
                                                            47
                                                                    * transaction ordering. One possible solution
    to mitigate this race
                                                                to mitigate this race
                                                                     * condition is to first reduce the spender's a
48
        * condition is to first reduce the spender's a
                                                            48
   llowance to 0 and set the
                                                                llowance to 0 and set the
49
        * desired value afterwards:
                                                            49
                                                                     * desired value afterwards:
        * https://github.com/ethereum/EIPs/issues/20#i
                                                                     * https://github.com/ethereum/EIPs/issues/20#i
   ssuecomment-263524729
                                                                ssuecomment-263524729
51
                                                            51
        * Emits an {Approval} event.
52
                                                            52
                                                                     * Emits an {Approval} event.
53
                                                            53
       function approve(address spender, uint256 amoun
                                                                    function approve(address spender, uint256 amoun
   t) external returns (bool);
                                                                t) external returns (bool):
55
                                                            55
56
                                                            56
       * @dev Moves `amount` tokens from `sender` to
                                                                    * @dev Moves `amount` tokens from `sender` to
57
    `recipient` using the
                                                                 `recipient` using the
        * allowance mechanism. `amount` is then deduct
                                                                     * allowance mechanism. `amount` is then deduct
                                                            58
   ed from the caller's
                                                                ed from the caller's
        * allowance.
                                                                     * allowance.
59
                                                            59
60
                                                            60
        * Returns a boolean value indicating whether t
                                                                     * Returns a boolean value indicating whether t
   he operation succeeded.
                                                                he operation succeeded.
62
                                                            62
63
        * Emits a {Transfer} event.
                                                            63
                                                                     * Emits a {Transfer} event.
64
                                                            64
65
       function transferFrom(address sender, address r
                                                            65
                                                                    function transferFrom(address sender, address r
   ecipient, uint256 amount) external returns (bool);
                                                                ecipient, uint256 amount) external returns (bool);
                                                            66
66
67
                                                            67
68
       * @dev Emitted when `value` tokens are moved f
                                                            68
                                                                    * @dev Emitted when `value` tokens are moved f
   rom one account (`from`) to
                                                                rom one account (`from`) to
       * another (`to`).
                                                                     * another (`to`).
69
                                                            69
70
                                                            70
71
        * Note that `value` may be zero.
                                                            71
                                                                     * Note that `value` may be zero.
72
                                                            72
                                                                     */
       event Transfer(address indexed from, address in
                                                                    event Transfer(address indexed from, address in
   dexed to, uint256 value);
                                                                dexed to, uint256 value);
74
                                                            74
75
                                                            75
                                                                    * @dev Emitted when the allowance of a `spende
        * @dev Emitted when the allowance of a `spende
76
                                                            76
   r' for an 'owner' is set by
                                                                r' for an 'owner' is set by
       * a call to {approve}. `value` is the new allo
                                                                    * a call to {approve}. `value` is the new allo
   wance.
                                                                wance.
78
       */
                                                            78
       event Approval(address indexed owner, address i
                                                                   event Approval(address indexed owner, address i
   ndexed spender, uint256 value);
                                                                ndexed spender, uint256 value);
                                                            80 }
80 }
82 library LowGasSafeMath {
                                                            82 library LowGasSafeMath {
      /// @notice Returns x + y, reverts if sum overf
                                                                    /// @notice Returns x + y, reverts if sum overf
   lows uint256
                                                                lows uint256
     /// @param x The augend
                                                                  /// @param x The augend
      /// @param y The addend
                                                                    /// @param y The addend
85
                                                            85
      /// @return z The sum of x and y
                                                                    /// @return z The sum of x and v
86
                                                            86
                                                                    function add(uint256 x, uint256 y) internal pur
      function add(uint256 x, uint256 y) internal pur
                                                            87
   e returns (uint256 z) {
                                                                e returns (uint256 z) {
88
          require((z = x + y) >= x);
                                                            88
                                                                        require((z = x + y) >= x);
89
                                                            89
90
                                                            90
       function add32(uint32 x, uint32 y) internal pur
                                                                    function add32(uint32 x, uint32 y) internal pur
                                                            91
   e returns (uint32 z) {
                                                                e returns (uint32 z) {
                                                                        require((z = x + y) >= x);
          require((z = x + y) >= x);
                                                            92
92
93
                                                            93
94
                                                            94
95
       /// @notice Returns x - y, reverts if underflow
                                                            95
                                                                    /// @notice Returns x - y, reverts if underflow
96
       /// @param x The minuend
                                                            96
                                                                    /// @param x The minuend
       /// @param y The subtrahend
97
                                                            97
                                                                    /// @param y The subtrahend
       /// @return z The difference of x and y
                                                                    /// @return z The difference of x and y
98
                                                            98
```

```
e returns (uint256 z) {
                                                                  e returns (uint256 z) {
 100
            require((z = x - y) \le x);
                                                              100
                                                                          require((z = x - y) <= x);
 101
                                                              101
 102
                                                              102
         function sub32(uint32 x, uint32 y) internal pur
                                                                      function sub32(uint32 x, uint32 y) internal pur
 103
                                                              103
     e returns (uint32 z) {
                                                                  e returns (uint32 z) {
 104
             require((z = x - y) \le x);
                                                              104
                                                                          require((z = x - y) <= x);
 105
                                                              105
                                                                      }
         }
                                                              106
         /// @notice Returns x * y, reverts if overflows
                                                                      /// @notice Returns x * y, reverts if overflows
         /// @param x The multiplicand
                                                                      /// @param x The multiplicand
                                                             108
 109
         /// @param y The multiplier
                                                              109
                                                                      /// @param y The multiplier
         /// @return z The product of x and y
                                                              110
                                                                      /// @return z The product of x and y
         function mul(uint256 x, uint256 y) internal pur
                                                                      function mul(uint256 x, uint256 y) internal pur
                                                             111
     e returns (uint256 z) {
                                                                  e returns (uint256 z) {
                                                                         require(x == 0 || (z = x * y) / x == y);
            require(x == 0 || (z = x * y) / x == y);
 112
                                                              112
                                                              113
 113
 114
 115
         /// @notice Returns x + y, reverts if overflows
                                                              115
                                                                      /// @notice Returns x + y, reverts if overflows
     or underflows
                                                                  or underflows
         /// @param x The augend
                                                              116
                                                                      /// @param x The augend
 117
         /// @param y The addend
                                                              117
                                                                      /// @param y The addend
         /// @return z The sum of x and y
                                                                      /// @return z The sum of x and y
 118
                                                              118
         function add(int256 x, int256 y) internal pure
                                                                      function add(int256 x, int256 y) internal pure
 119
                                                              119
      returns (int256 z) {
                                                                   returns (int256 z) {
 120
           require((z = x + y) >= x == (y >= 0));
                                                              120
                                                                        require((z = x + y) >= x == (y >= 0));
 121
                                                              121
         /// @notice Returns x - y, reverts if overflows
                                                              123
                                                                      /// @notice Returns x - y, reverts if overflows
     or underflows
                                                                  or underflows
         /// @param x The minuend
                                                              124
                                                                     /// @param x The minuend
 124
                                                                     /// @param y The subtrahend
 125
         /// @param y The subtrahend
                                                              125
126
         /// @return z The difference of x and y
                                                                      /// @return z The difference of x and y
                                                              126
 127
         function sub(int256 x, int256 y) internal pure
                                                                     function sub(int256 x, int256 y) internal pure
                                                             127
      returns (int256 z) {
                                                                   returns (int256 z) {
           require((z = x - y) \le x == (y >= 0));
                                                              128
                                                                        require((z = x - y) \le x == (y >= 0));
 129
                                                              129
 130
                                                              130
         function div(uint256 x, uint256 y) internal pur
                                                                      function div(uint256 x, uint256 y) internal pur
     e returns(uint256 z){
                                                                  e returns(uint256 z){
 132
         require(y > 0);
                                                              132
                                                                        require(y > 0);
             z=x/y;
                                                              133
                                                                          z=x/y;
 134
                                                              134
 135 }
                                                              135 }
 136
                                                              136
 137 /**
                                                             137 /**
                                                             138 * @dev Collection of functions related to the addr
     * @dev Collection of functions related to the addr
 138
     ess type
                                                                  ess type
 139 */
                                                              139 */
 140 library Address {
                                                              140 library Address {
                                                              141
          * @dev Returns true if `account` is a contrac
                                                                       * @dev Returns true if `account` is a contrac
     t.
 143
                                                              143
          * [IMPORTANT]
                                                                       * [IMPORTANT]
 144
                                                              144
 145
                                                              145
                                                                       ^{\ast} It is unsafe to assume that an address for w
          ^{\ast} It is unsafe to assume that an address for w
     hich this function returns
                                                                  hich this function returns
          * false is an externally-owned account (EOA) a
                                                                      * false is an externally-owned account (EOA) a
                                                                  nd not a contract.
     nd not a contract.
 148
                                                              148
          ^{\star} Among others, 'isContract' will return false
                                                                       * Among others, `isContract` will return false
 149
                                                              149
     for the following
                                                                  for the following
 150
          * types of addresses:
                                                              150
                                                                      * types of addresses:
 151
                                                              151
          * - an externally-owned account
                                                              152
                                                                       * - an externally-owned account
          * - a contract in construction
                                                                       * - a contract in construction
 153
                                                              153
```

function sub(uint256 x, uint256 y) internal pur

function sub(uint256 x, uint256 y) internal pur

```
* - an address where a contract will be creat
                                                                      * - an address where a contract will be creat
                                                                  ed
    ed
         ^{\star}\,\, - an address where a contract lived, but wa
                                                                       * - an address where a contract lived, but wa
                                                             155
    s destroyed
                                                                  s destroved
156
         * ====
                                                             156
                                                                       * ====
                                                                       */
157
                                                             157
        function isContract(address account) internal v
                                                                      function isContract(address account) internal v
    iew returns (bool) {
                                                                  iew returns (bool) {
            // This method relies in extcodesize, which
                                                                         // This method relies in extcodesize, which
    returns 0 for contracts in
                                                                  returns 0 for contracts in
160
            // construction, since the code is only sto
                                                             160
                                                                          // construction, since the code is only sto
    red at the end of the
                                                                  red at the end of the
            // constructor execution.
                                                                          // constructor execution.
161
                                                             161
162
                                                             162
163
            uint256 size;
                                                             163
                                                                          uint256 size;
164
            // solhint-disable-next-line no-inline-asse
                                                                          // solhint-disable-next-line no-inline-asse
    mblv
                                                                  mblv
            assemblv { size := extcodesize(account) }
                                                                          assemblv { size := extcodesize(account) }
165
                                                             165
166
            return size > 0;
                                                                          return size > 0;
167
                                                             167
169
                                                             169
170
         * @dev Replacement for Solidity's `transfer`:
                                                                       * @dev Replacement for Solidity's `transfer`:
     sends `amount` wei to
                                                                   sends `amount` wei to
         * `recipient`, forwarding all available gas an
                                                                       * `recipient`, forwarding all available gas an
    d reverting on errors.
                                                                  d reverting on errors.
172
                                                             172
         * https://eips.ethereum.org/EIPS/eip-1884[EIP1
                                                                       * https://eips.ethereum.org/EIPS/eip-1884[EIP1
    884] increases the gas cost
                                                                  884] increases the gas cost
         * of certain opcodes, possibly making contract
                                                                       * of certain opcodes, possibly making contract
    s go over the 2300 gas limit
                                                                  s go over the 2300 gas limit
175
         * imposed by `transfer`, making them unable to
                                                                       * imposed by `transfer`, making them unable to
                                                             175
    receive funds via
                                                                  receive funds via
176
         * `transfer`. {sendValue} removes this limitat
                                                             176
                                                                       * `transfer`. {sendValue} removes this limitat
    ion.
                                                                  ion.
177
                                                             177
178
         * https://diligence.consensys.net/posts/2019/0
                                                                       * https://diligence.consensys.net/posts/2019/0
    9/stop-using-soliditys-transfer-now/[Learn more].
                                                                  9/stop-using-soliditys-transfer-now/[Learn more].
179
                                                             179
         ^{\star} IMPORTANT: because control is transferred to
                                                                       ^{\star} IMPORTANT: because control is transferred to
180
                                                             180
     `recipient`, care must be
                                                                  `recipient`, care must be
181
         * taken to not create reentrancy vulnerabiliti
                                                             181
                                                                      * taken to not create reentrancy vulnerabiliti
    es. Consider using
                                                                  es. Consider using
182
         * {ReentrancyGuard} or the
                                                             182
                                                                       * {ReentrancyGuard} or the
183
          * https://solidity.readthedocs.io/en/v0.5.11/s
                                                             183
                                                                       * https://solidity.readthedocs.io/en/v0.5.11/s
    ecurity-considerations.html#use-the-checks-effects-
                                                                  ecurity-considerations.html#use-the-checks-effects-
    interactions-pattern[checks-effects-interactions pa
                                                                  interactions-pattern[checks-effects-interactions pa
    ttern1.
                                                                  ttern1.
184
                                                             184
        function sendValue(address payable recipient, u
                                                                      function sendValue(address payable recipient, u
    int256 amount) internal {
                                                                  int256 amount) internal {
           require(address(this).balance >= amount, "A
                                                                         require(address(this).balance >= amount, "A
186
    ddress: insufficient balance"):
                                                                  ddress: insufficient balance"):
187
                                                             187
            // solhint-disable-next-line avoid-low-leve
188
                                                             188
                                                                          // solhint-disable-next-line avoid-low-leve
    l-calls, avoid-call-value
                                                                  l-calls, avoid-call-value
            (bool success, ) = recipient.call{ value: a
                                                                          (bool success, ) = recipient.call{ value: a
189
                                                             189
    mount }("");
                                                                  mount }("");
            require(success, "Address: unable to send v
                                                                          require(success, "Address: unable to send v
190
    alue, recipient may have reverted");
                                                                  alue, recipient may have reverted");
191
        }
                                                             191
                                                                      }
193
                                                             193
         * @dev Performs a Solidity function call using
                                                                       * @dev Performs a Solidity function call using
10/
                                                             194
    a low level `call`. A
                                                                  a low level `call`. A
         * plain`call` is an unsafe replacement for a f
                                                                       * plain`call` is an unsafe replacement for a f
    unction call: use this
                                                                  unction call: use this
         * function instead.
                                                                       * function instead.
196
                                                             196
```

154

```
* If `target` reverts with a revert reason, it
                                                             198
                                                                      * If `target` reverts with a revert reason, it
198
    is bubbled up by this
                                                                is bubbled up by this
199
         * function (like regular Solidity function cal
                                                             199
                                                                      * function (like regular Solidity function cal
                                                                 ls).
    ls).
                                                             200
200
         * Returns the raw returned data. To convert to
                                                                      * Returns the raw returned data. To convert to
    the expected return value,
                                                                 the expected return value,
         * use https://solidity.readthedocs.io/en/lates
                                                                      * use https://solidity.readthedocs.io/en/lates
    t/units-and-global-variables.html?highlight=abi.dec
                                                                 t/units-and-global-variables.html?highlight=abi.dec
    ode#abi-encoding-and-decoding-functions[`abi.decode
                                                                 ode#abi-encoding-and-decoding-functions[`abi.decode
     `].
                                                                  `].
203
                                                             203
         * Requirements:
                                                             204
                                                                      * Requirements:
204
                                                             205
205
                                                                      \star - `target` must be a contract.
         * - `target` must be a contract.
206
                                                             206
         * - calling `target` with `data` must not reve
                                                                       * - calling `target` with `data` must not reve
    rt.
                                                                 rt.
                                                             208
209
         * _Available since v3.1._
                                                             209
                                                                      * _Available since v3.1._
210
        function functionCall(address target, bytes mem
                                                                     function functionCall(address target, bytes mem
211
    ory data) internal returns (bytes memory) {
                                                                 ory data) internal returns (bytes memory) {
        return functionCall(target, data, "Address: l
                                                                      return functionCall(target, data, "Address: l
    ow-level call failed");
                                                                 ow-level call failed");
213
        }
                                                             213
                                                                     }
214
                                                             214
215
                                                             215
         * @dev Same as {xref-Address-functionCall-addr
                                                                      * @dev Same as {xref-Address-functionCall-addr
    ess-bytes-}[`functionCall`], but with
                                                                 ess-bytes-}[`functionCall`], but with
         * `errorMessage` as a fallback revert reason w
                                                                      * `errorMessage` as a fallback revert reason w
217
                                                             217
                                                                 hen `target` reverts.
    hen `target` reverts.
218
                                                             218
         * _Available since v3.1._
                                                                      * _Available since v3.1._
219
                                                             219
220
                                                             220
        function functionCall(address target, bytes mem
                                                                     function functionCall(address target, bytes mem
    ory data, string memory errorMessage) internal retu
                                                                 ory data, string memory errorMessage) internal retu
    rns (bytes memory) {
                                                                 rns (bytes memory) {
          return functionCallWithValue(target, data,
                                                                        return functionCallWithValue(target, data,
    0, errorMessage);
                                                                 0, errorMessage);
223
        }
                                                             223
                                                                     }
224
                                                             224
                                                             225
        * @dev Same as {xref-Address-functionCall-addr
                                                                      * @dev Same as {xref-Address-functionCall-addr
    ess-bytes-}[`functionCall`],
                                                                 ess-bytes-}[`functionCall`],
         * but also transferring `value` wei to `target
                                                                      * but also transferring `value` wei to `target
227
                                                             227
228
                                                             228
         * Requirements:
229
                                                             229
                                                                      * Requirements:
         * - the calling contract must have an ETH bala
                                                                      * - the calling contract must have an ETH bala
    nce of at least `value`.
                                                                 nce of at least `value`.
         * - the called Solidity function must be `paya
                                                                      * - the called Solidity function must be `paya
    hle`.
                                                                 hle`.
          * _Available since v3.1._
                                                             234
                                                                       * _Available since v3.1._
234
        function functionCallWithValue(address target,
                                                                     function functionCallWithValue(address target,
     bytes memory data, uint256 value) internal returns
                                                                  bytes memory data, uint256 value) internal returns
    (bytes memory) {
                                                                  (bytes memory) {
            return functionCallWithValue(target, data,
                                                                         return functionCallWithValue(target, data,
237
                                                             237
     value, "Address: low-level call with value faile
                                                                  value, "Address: low-level call with value faile
    d");
                                                                 d");
238
        }
                                                             238
                                                                     }
239
                                                             239
```

197

```
alue-address-bytes-uint256-}[`functionCallWithValue
                                                                  alue-address-bytes-uint256-}[`functionCallWithValue
     `1. but
                                                                   `1, but
         * with `errorMessage` as a fallback revert rea
242
                                                              242
                                                                       * with `errorMessage` as a fallback revert rea
    son when `target` reverts.
                                                                  son when `target` reverts.
243
                                                              243
         * _Available since v3.1._
                                                                       * _Available since v3.1._
245
246
        function functionCallWithValue(address target,
                                                              246
                                                                      function functionCallWithValue(address target,
     bytes memory data, uint256 value, string memory er
                                                                   bytes memory data, uint256 value, string memory er
    rorMessage) internal returns (bytes memory) {
                                                                  rorMessage) internal returns (bytes memory) {
247
            require(address(this).balance >= value, "Ad
                                                                          require(address(this).balance >= value, "Ad
    dress: insufficient balance for call");
                                                                  dress: insufficient balance for call");
            return _functionCallWithValue(target, data,
                                                                          return _functionCallWithValue(target, data,
    value, errorMessage);
                                                                  value, errorMessage);
249
        }
                                                              249
                                                                      }
        function functionCallWithValue(address target,
                                                                      function functionCallWithValue(address target,
251
                                                              251
    bytes memory data, uint256 weiValue, string memory
                                                                  bytes memory data, uint256 weiValue, string memory
     errorMessage) private returns (bytes memory) {
                                                                   errorMessage) private returns (bytes memory) {
            require(isContract(target), "Address: call
                                                                          require(isContract(target), "Address: call
     to non-contract");
                                                                   to non-contract");
253
                                                              253
            // solhint-disable-next-line avoid-low-leve
                                                              254
                                                                          // solhint-disable-next-line avoid-low-leve
254
                                                                  1-calls
    1-calls
             (bool success, bytes memory returndata) = t
                                                              255
                                                                          (bool success, bytes memory returndata) = t
    arget.call{ value: weiValue }(data);
                                                                  arget.call{ value: weiValue }(data);
256
            if (success) {
                                                              256
                                                                          if (success) {
                return returndata;
                                                                               return returndata;
                                                              258
            } else {
                                                                          } else {
                // Look for revert reason and bubble it
                                                                               // Look for revert reason and bubble it
    up if present
                                                                  up if present
                if (returndata.length > 0) {
                                                              260
                                                                              if (returndata.length > 0) {
261
                     // The easiest way to bubble the re
                                                              261
                                                                                   // The easiest way to bubble the re
    vert reason is using memory via assembly
                                                                  vert reason is using memory via assembly
262
                                                              262
263
                     // solhint-disable-next-line no-inl
                                                              263
                                                                                   // solhint-disable-next-line no-inl
    ine-assembly
                                                                  ine-assembly
264
                     assembly {
                                                              264
                                                                                   assembly {
265
                         let returndata size := mload(re
                                                              265
                                                                                       let returndata size := mload(re
    turndata)
                                                                  turndata)
266
                         revert(add(32, returndata), ret
                                                              266
                                                                                       revert(add(32, returndata), ret
    urndata size)
                                                                  urndata size)
                                                                                   }
                     }
268
                                                              268
                } else {
                                                                              } else {
269
                                                              269
                     revert(errorMessage);
                                                                                   revert(errorMessage);
                }
                                                                               }
271
            }
                                                              271
                                                                          }
272
                                                              272
        }
                                                                      }
                                                              273 }
273 }
274
                                                              274
                                                              275 /**
     * @dev Implementation of the {IERC20} interface.
                                                              276 * @dev Implementation of the {IERC20} interface.
276
277
     * This implementation is agnostic to the way token
                                                              ^{\rm 278} ^{\,\,\star} This implementation is agnostic to the way token
    s are created. This means
                                                                  s are created. This means
     * that a supply mechanism has to be added in a der
                                                              279 * that a supply mechanism has to be added in a der
    ived contract using {_mint}.
                                                                  ived contract using {_mint}.
280
     * For a generic mechanism see {ERC20PresetMinterPa
                                                              280
                                                                   * For a generic mechanism see {ERC20PresetMinterPa
    user}.
                                                                  user}.
281
                                                              281
                                                                   * TIP: For a detailed writeup see our guide
     * TIP: For a detailed writeup see our quide
                                                              282
282
     * https://forum.zeppelin.solutions/t/how-to-implem
                                                              283 * https://forum.zeppelin.solutions/t/how-to-implem
    ent-erc20-supply-mechanisms/226[How
                                                                  ent-erc20-supply-mechanisms/226[How
^{\rm 284} ^{\rm *} to implement supply mechanisms].
                                                              284 * to implement supply mechanisms].
     * We have followed general OpenZeppelin guideline
                                                              286 * We have followed general OpenZeppelin guideline
    s: functions revert instead
                                                                  s: functions revert instead
```

* @dev Same as {xref-Address-functionCallWithV

241

* @dev Same as {xref-Address-functionCallWithV

```
287 * of returning `false` on failure. This behavior i
                                                            287 * of returning `false` on failure. This behavior i
    s nonetheless conventional
                                                                 s nonetheless conventional
288
     * and does not conflict with the expectations of E
                                                             288
                                                                  * and does not conflict with the expectations of E
    RC20 applications.
                                                                 RC20 applications.
289 *
                                                             289 *
290 * Additionally, an {Approval} event is emitted on
                                                             290 * Additionally, an {Approval} event is emitted on
     calls to {transferFrom}.
                                                                  calls to {transferFrom}.
291 * This allows applications to reconstruct the allo
                                                             291 * This allows applications to reconstruct the allo
    wance for all accounts just
                                                                 wance for all accounts just
^{292} * by listening to said events. Other implementatio
                                                             ^{292} * by listening to said events. Other implementatio
    ns of the EIP may not emit
                                                                 ns of the EIP may not emit
293 * these events, as it isn't required by the specif
                                                             293 * these events, as it isn't required by the specif
                                                                 ication.
294
                                                             294 *
     * Finally, the non-standard {decreaseAllowance} an
                                                             295 * Finally, the non-standard {decreaseAllowance} an
    d {increaseAllowance}
                                                                 d {increaseAllowance}
296 * functions have been added to mitigate the well-k
                                                             296 * functions have been added to mitigate the well-k
    nown issues around setting
                                                                 nown issues around setting
297 * allowances. See {IERC20-approve}.
                                                             297 * allowances. See {IERC20-approve}.
298 */
                                                             298 */
299 contract ERC20 is IERC20 {
                                                             299 contract ERC20 is IERC20 {
300
        using LowGasSafeMath for uint256;
                                                             300
                                                                     using LowGasSafeMath for uint256;
301
                                                             301
302
        mapping (address => uint256) private balances:
                                                             302
                                                                     mapping (address => uint256) private balances:
303
                                                             303
        mapping (address => mapping (address => uint25
                                                                     mapping (address => mapping (address => uint25
304
                                                             304
    private _allowances;
                                                                 private _allowances;
305
                                                             305
306
        uint256 private _totalSupply;
                                                             306
                                                                     uint256 private _totalSupply;
                                                             307
307
308
        string private _name;
                                                             308
                                                                     string private _name;
309
        string private _symbol;
                                                             309
                                                                     string private _symbol;
                                                                     uint8 public immutable decimals;
        uint8 public immutable decimals;
310
                                                             310
311
                                                             311
312
                                                             312
         * @dev Sets the values for {name} and {symbo
313
                                                             313
                                                                      * @dev Sets the values for {name} and {symbo
    l}, initializes {decimals} with
                                                                 l}, initializes {decimals} with
314
         * a default value of 18.
                                                             314
                                                                      * a default value of 18.
315
                                                             315
         ^{\star} To select a different value for {decimals},
                                                                      * To select a different value for {decimals},
316
                                                             316
     use {_setupDecimals}.
                                                                  use {_setupDecimals}.
317
                                                             317
         * All three of these values are immutable: the
                                                                      * All three of these values are immutable: the
    y can only be set once during
                                                                 y can only be set once during
         * construction.
                                                                      * construction.
319
320
                                                             320
321
        constructor (string memory name, string memory
                                                                     constructor (string memory name, string memory
                                                             321
     symbol) {
                                                                   symbol) {
322
           _name = name;
                                                             322
                                                                         _name = name;
            _symbol = symbol;
                                                             323
                                                                         _symbol = symbol;
323
            decimals = 18;
                                                             324
                                                                         decimals = 18;
324
325
                                                             325
                                                                     }
        }
326
                                                             326
                                                                     /**
        /**
327
                                                             327
         ^{\ast} @dev Returns the name of the token.
                                                                      * @dev Returns the name of the token.
328
                                                             328
329
                                                             329
        function name() public view returns (string mem
                                                                     function name() public view returns (string mem
    ory) {
                                                                 ory) {
331
            return _name;
                                                             331
                                                                         return _name;
332
                                                             332
        }
                                                                     }
                                                             333
333
334
                                                             334
                                                                      ^{\star} @dev Returns the symbol of the token, usuall
         * @dev Returns the symbol of the token, usuall
335
                                                             335
    y a shorter version of the
                                                                 y a shorter version of the
                                                             336
                                                                      * name.
336
         * name.
337
                                                                      */
        function symbol() public view returns (string m
                                                                     function symbol() public view returns (string m
    emory) {
                                                                 emory) {
339
                                                             339
          return _symbol;
                                                                       return symbol:
```

```
341
                                                              341
                                                              342
342
         * @dev See {IERC20-totalSupply}.
                                                                        * @dev See {IERC20-totalSupply}.
343
                                                              343
344
                                                              344
        function totalSupply() public view override ret
                                                                       function totalSupply() public view override ret
345
                                                              345
    urns (uint256) {
                                                                   urns (uint256) {
346
            return _totalSupply;
                                                              346
                                                                           return _totalSupply;
347
                                                              347
        }
                                                              348
348
349
                                                              349
         * @dev See {IERC20-balanceOf}.
                                                                        * @dev See {IERC20-balanceOf}.
350
                                                              350
351
                                                              351
352
        function balanceOf(address account) public view
                                                                       function balanceOf(address account) public view
    override returns (uint256) {
                                                                   override returns (uint256) {
353
            return _balances[account];
                                                              353
                                                                           return _balances[account];
                                                              354
354
        }
                                                                       }
355
                                                              355
356
                                                              356
         * @dev See {IERC20-transfer}.
                                                                        * @dev See {IERC20-transfer}.
357
                                                              357
                                                              358
         * Requirements:
                                                                        * Requirements:
361
          * - `recipient` cannot be the zero address.
                                                              361
                                                                        * - `recipient` cannot be the zero address.
         * - the caller must have a balance of at least
                                                                        * - the caller must have a balance of at least
362
                                                              362
     `amount`.
                                                                    `amount`.
363
                                                              363
        function transfer(address recipient, uint256 am
                                                              364
                                                                       function transfer(address recipient, uint256 am
    ount) public virtual override returns (bool) {
                                                                   ount) public virtual override returns (bool) {
            _transfer(msg.sender, recipient, amount);
                                                                           _transfer(msg.sender, recipient, amount);
366
            return true;
                                                              366
                                                                           return true;
367
                                                              367
        }
                                                                       }
368
                                                              368
369
                                                              369
         * @dev See {IERC20-allowance}.
                                                                        * @dev See {IERC20-allowance}.
370
                                                              370
371
                                                              371
                                                                       function allowance(address owner, address spend
        function allowance(address owner, address spend
    er) public view virtual override returns (uint256)
                                                                   er) public view virtual override returns (uint256)
373
            return _allowances[owner][spender];
                                                              373
                                                                           return _allowances[owner][spender];
                                                              374
374
        }
                                                                       }
375
                                                              375
376
                                                              376
         * @dev See {IERC20-approve}.
                                                                        * @dev See {IERC20-approve}.
                                                              377
378
         * Requirements:
                                                                        * Requirements:
379
                                                              379
380
                                                              380
          ^{\star} - `spender` cannot be the zero address.
                                                                        \mbox{\scriptsize *} - `spender` cannot be the zero address.
381
                                                              381
382
                                                              382
         function approve(address spender, uint256 amoun
                                                                       function approve(address spender, uint256 amoun
    t) public virtual override returns (bool) {
                                                                   t) public virtual override returns (bool) {
            _approve(msg.sender, spender, amount);
                                                              384
                                                                           _approve(msg.sender, spender, amount);
385
            return true;
                                                              385
                                                                           return true;
                                                              386
386
        }
                                                                       }
388
                                                              388
         * @dev See {IERC20-transferFrom}.
                                                                        * @dev See {IERC20-transferFrom}.
389
                                                              389
390
                                                               390
          * Emits an {Approval} event indicating the upd
                                                                        * Emits an {Approval} event indicating the upd
    ated allowance. This is not
                                                                   ated allowance. This is not
                                                                        ^{\ast} required by the EIP. See the note at the beg
392
          * required by the EIP. See the note at the beg
                                                              392
    inning of {ERC20};
                                                                   inning of {ERC20};
393
                                                              393
394
         * Requirements:
                                                              394
                                                                        * Requirements:
          ^{\star} - `sender` and `recipient` cannot be the zer
                                                                        * - `sender` and `recipient` cannot be the zer
                                                                   o address.
         * - `sender` must have a balance of at least `
                                                                        * - `sender` must have a balance of at least `
    amount`.
                                                                   amount`.
```

}

340

}

```
r``'s tokens of at least
                                                                   r``'s tokens of at least
         * `amount`.
                                                                         * `amount`.
398
                                                              398
399
                                                              399
400
        function transferFrom(address sender, address r
                                                              400
                                                                       function transferFrom(address sender, address r
    ecipient, uint256 amount) public virtual override r
                                                                   ecipient, uint256 amount) public virtual override r
401
            _transfer(sender, recipient, amount);
                                                              401
                                                                           _transfer(sender, recipient, amount);
402
             _approve(sender, msg.sender, _allowances[se
                                                                           _approve(sender, msg.sender, _allowances[se
                                                              402
    nder][msg.sender].sub(amount));
                                                                   nder][msg.sender].sub(amount));
403
            return true;
                                                              403
                                                                           return true;
404
                                                              404
        }
                                                              405
405
406
                                                               406
                                                                        * @dev Atomically increases the allowance gran
         * @dev Atomically increases the allowance gran
    ted to `spender` by the caller.
                                                                   ted to `spender` by the caller.
408
                                                              408
          ^{\star} This is an alternative to {approve} that can
                                                                        ^{\star} This is an alternative to {approve} that can
409
                                                              409
    be used as a mitigation for
                                                                   be used as a mitigation for
          * problems described in {IERC20-approve}.
                                                                        * problems described in {IERC20-approve}.
410
                                                              410
411
                                                               411
          * Emits an {Approval} event indicating the upd
                                                                         * Emits an {Approval} event indicating the upd
    ated allowance.
                                                                   ated allowance.
413
                                                              413
         * Requirements:
                                                                        * Requirements:
414
                                                              414
415
                                                              415
          * - `spender` cannot be the zero address.
                                                                        * - `spender` cannot be the zero address.
416
                                                              416
417
                                                              417
418
        function increaseAllowance(address spender, uin
                                                                       function increaseAllowance(address spender, uin
    t256 addedValue) public virtual returns (bool) {
                                                                   t256 addedValue) public virtual returns (bool) {
            _approve(msg.sender, spender, _allowances[m
                                                                           _approve(msg.sender, spender, _allowances[m
419
                                                              419
    sg.sender][spender].add(addedValue));
                                                                   sg.sender][spender].add(addedValue));
420
            return true;
                                                              420
                                                                           return true;
421
                                                              121
422
                                                              422
423
         * @dev Atomically decreases the allowance gran
424
                                                                        * @dev Atomically decreases the allowance gran
    ted to `spender` by the caller.
                                                                   ted to `spender` by the caller.
425
                                                              425
                                                                        ^{\star} This is an alternative to {approve} that can \,
          ^{\ast} This is an alternative to {approve} that can
426
                                                              426
    be used as a mitigation for
                                                                   be used as a mitigation for
427
         * problems described in {IERC20-approve}.
                                                               427
                                                                        * problems described in {IERC20-approve}.
429
          * Emits an {Approval} event indicating the upd
                                                                        * Emits an {Approval} event indicating the upd
    ated allowance.
                                                                   ated allowance.
                                                              430
430
         * Requirements:
                                                                        * Requirements:
                                                              431
431
432
                                                              432
         ^{\star} - `spender` cannot be the zero address.
                                                              433
                                                                        * - `spender` cannot be the zero address.
433
         * - `spender` must have allowance for the call
                                                                        * - `spender` must have allowance for the call
    er of at least
                                                                   er of at least
          * `subtractedValue`.
                                                                        * `subtractedValue`.
435
                                                              435
                                                                        * /
436
                                                              436
        function decreaseAllowance(address spender, uin
                                                                       function decreaseAllowance(address spender, uin
437
                                                              437
                                                                   t256 subtractedValue) public virtual returns (bool)
    t256 subtractedValue) public virtual returns (bool)
    {
                                                                   {
             _approve(msg.sender, spender, _allowances[m
                                                                           _approve(msg.sender, spender, _allowances[m
438
                                                               438
    sg.sender][spender].sub(subtractedValue));
                                                                   sg.sender][spender].sub(subtractedValue));
            return true;
                                                              439
                                                                           return true;
439
440
        }
                                                              440
                                                                       }
441
                                                               441
442
                                                              442
         * @dev Moves tokens `amount` from `sender` to
                                                                        * @dev Moves tokens `amount` from `sender` to
443
      `recipient`.
                                                                     `recipient`.
444
                                                               444
          * This is internal function is equivalent to
                                                                         * This is internal function is equivalent to
     {transfer}, and can be used to
                                                                    {transfer}, and can be used to
```

* - the caller must have allowance for ``sende

397

* - the caller must have allowance for ``sende

```
* e.g. implement automatic token fees, slashin
                                                              446
                                                                        * e.g. implement automatic token fees, slashin
                                                                   g mechanisms, etc.
     a mechanisms, etc.
 447
                                                               447
          * Emits a {Transfer} event.
                                                                        * Emits a {Transfer} event.
 448
                                                               448
 449
                                                               449
          * Requirements:
                                                                        * Requirements:
 450
                                                               450
 451
          * - `sender` cannot be the zero address.
                                                               452
                                                                         * - `sender` cannot be the zero address.
           * - `recipient` cannot be the zero address.
                                                               453
                                                                         * - `recipient` cannot be the zero address.
 453
           * - `sender` must have a balance of at least `
                                                                        * - `sender` must have a balance of at least `
 454
                                                               454
     amount`.
                                                                   amount`.
 455
                                                               455
          function _transfer(address sender, address reci
                                                                       function _transfer(address sender, address reci
                                                                   pient, uint256 amount) internal virtual {
     pient, uint256 amount) internal virtual {
 457
             require(sender != address(0), "ERC20: trans
                                                                           require(sender != address(0), "ERC20: trans
     fer from the zero address");
                                                                   fer from the zero address");
             require(recipient != address(0), "ERC20: tr
                                                               458
                                                                           require(recipient != address(0), "ERC20: tr
 458
     ansfer to the zero address");
                                                                   ansfer to the zero address");
 459
                                                               459
             _beforeTokenTransfer(sender, recipient, amo
                                                               460
                                                                           _beforeTokenTransfer(sender, recipient, amo
 460
     unt);
                                                                   unt);
 461
                                                               461
 462
             _balances[sender] = _balances[sender].sub(a
                                                               462
                                                                            _balances[sender] = _balances[sender].sub(a
     mount);
                                                                   mount);
              _balances[recipient] = _balances[recipien
                                                                            _balances[recipient] = _balances[recipien
 463
                                                               463
     tl.add(amount);
                                                                   tl.add(amount);
 464
             emit Transfer(sender, recipient, amount);
                                                               464
                                                                            emit Transfer(sender, recipient, amount);
 465
                                                               465
 466
                                                               466
         /** @dev Creates `amount` tokens and assigns th
                                                                        /** @dev Creates `amount` tokens and assigns th
     em to `account`, increasing
                                                                   em to `account`, increasing
          * the total supply.
                                                                         * the total supply.
 468
                                                               468
 469
                                                               469
           * Emits a {Transfer} event with `from` set to
                                                                         * Emits a {Transfer} event with `from` set to
470
                                                               470
      the zero address.
                                                                     the zero address.
 471
                                                               471
          * Requirements
 472
                                                               472
                                                                         * Requirements
                                                               473
 473
           * - `to` cannot be the zero address.
 474
                                                               474
                                                                         * - `to` cannot be the zero address.
                                                                        */
 475
                                                               475
 476
         function _mint(address account, uint256 amount)
                                                               476
                                                                       function _mint(address account, uint256 amount)
     internal virtual {
                                                                   internal virtual {
             require(account != address(0), "ERC20: mint
                                                                           require(account != address(0), "ERC20: mint
 477
                                                               477
     to the zero address");
                                                                    to the zero address");
 478
                                                               478
             _beforeTokenTransfer(address(0), account, a
                                                               479
                                                                            _beforeTokenTransfer(address(0), account, a
 479
     mount):
                                                                   mount):
                                                               480
 481
              _totalSupply = _totalSupply.add(amount);
                                                               481
                                                                            _totalSupply = _totalSupply.add(amount);
             _balances[account] = _balances[account].add
                                                                           _balances[account] = _balances[account].add
 482
                                                                    (amount);
      (amount);
 483
             emit Transfer(address(0), account, amount);
                                                               483
                                                                            emit Transfer(address(0), account, amount);
 484
                                                               484
         }
                                                                       }
 485
                                                               485
 486
                                                               486
          * @dev Destroys `amount` tokens from `account
                                                                        ^{\star} @dev Destroys `amount` tokens from `account
 487
                                                               487
       , reducing the
                                                                     , reducing the
          * total supply.
                                                                        * total supply.
 488
                                                               488
           * Emits a {Transfer} event with `to` set to th
                                                                         * Emits a {Transfer} event with `to` set to th
 490
                                                               490
     e zero address.
                                                                   e zero address.
 491
                                                               491
          * Requirements
                                                                        * Requirements
 492
                                                               492
 493
                                                               493
 494
           * - `account` cannot be the zero address.
                                                               494
                                                                        * - `account` cannot be the zero address.
           * - `account` must have at least `amount` toke
                                                                         * - `account` must have at least `amount` toke
                                                                   ns.
     ns.
                                                                         */
 496
                                                               496
```

```
internal virtual {
                                                                  internal virtual {
           require(account != address(0), "ERC20: burn
                                                                          require(account != address(0), "ERC20: burn
498
                                                              498
    from the zero address");
                                                                   from the zero address");
499
                                                              499
            _beforeTokenTransfer(account, address(0), a
                                                                          _beforeTokenTransfer(account, address(0), a
                                                              500
    mount);
                                                                   mount);
501
                                                              501
502
            _balances[account] = _balances[account].sub
                                                              502
                                                                           _balances[account] = _balances[account].sub
    (amount);
                                                                   (amount);
            _totalSupply = _totalSupply.sub(amount);
                                                                           _totalSupply = _totalSupply.sub(amount);
504
            emit Transfer(account, address(0), amount);
                                                              504
                                                                           emit Transfer(account, address(0), amount);
505
                                                              505
                                                              506
507
         * @dev Sets `amount` as the allowance of `spen
                                                                        * @dev Sets `amount` as the allowance of `spen
    der' over the 'owner' s tokens.
                                                                   der' over the 'owner' s tokens.
509
                                                              509
         ^{\star} This internal function is equivalent to 'app \,
                                                                        ^{\star} This internal function is equivalent to `app
510
                                                              510
    rove`, and can be used to
                                                                   rove`, and can be used to
         * e.g. set automatic allowances for certain su
                                                                       * e.g. set automatic allowances for certain su
    bsystems, etc.
                                                                   bsystems, etc.
512
                                                              512
         * Emits an {Approval} event.
                                                              513
                                                                        * Emits an {Approval} event.
513
514
                                                              514
         * Requirements:
                                                                       * Requirements:
515
                                                              515
516
                                                              516
         ^{\star} - 'owner' cannot be the zero address.
                                                                        * - `owner` cannot be the zero address.
517
                                                              517
         * - `spender` cannot be the zero address.
                                                                        * - `spender` cannot be the zero address.
519
520
        function approve(address owner, address spende
                                                                       function approve(address owner, address spende
    r, uint256 amount) internal virtual {
                                                                   r, uint256 amount) internal virtual {
            require(owner != address(0), "ERC20: approv
                                                                          require(owner != address(0), "ERC20: approv
521
    e from the zero address");
                                                                   e from the zero address");
522
            require(spender != address(0), "ERC20: appr
                                                                          require(spender != address(0), "ERC20: appr
                                                                   ove to the zero address");
    ove to the zero address");
523
                                                              523
            _allowances[owner][spender] = amount;
                                                              524
                                                                           _allowances[owner][spender] = amount;
524
            emit Approval(owner, spender, amount);
525
                                                              525
                                                                          emit Approval(owner, spender, amount);
526
        }
                                                              526
                                                                       }
528
                                                              528
         * @dev Hook that is called before any transfer
                                                                       * @dev Hook that is called before any transfer
    of tokens. This includes
                                                                   of tokens. This includes
         * minting and burning.
                                                                        * minting and burning.
530
531
                                                              531
         * Calling conditions:
                                                                        * Calling conditions:
                                                              532
532
533
                                                              533
         ^{\star} - when 'from' and 'to' are both non-zero, 'a
                                                                        ^{\star} - when 'from' and 'to' are both non-zero, 'a
    mount` of ``from``'s tokens
                                                                   mount` of ``from``'s tokens
           will be to transferred to `to`.
                                                                         will be to transferred to `to`.
535
                                                              535
          * - when `from` is zero, `amount` tokens will
                                                                        * - when `from` is zero, `amount` tokens will
     be minted for `to`.
                                                                   be minted for `to`.
         * - when `to` is zero, `amount` of ``from``'s
                                                                       * - when `to` is zero, `amount` of ``from``'s
537
                                                              537
     tokens will be burned.
                                                                    tokens will be burned.
         ^{\star} - `from` and `to` are never both zero.
                                                                       ^{\star} - `from` and `to` are never both zero.
538
                                                              538
539
                                                              539
         * To learn more about hooks, head to xref:ROO
                                                                       * To learn more about hooks, head to xref:R00
    T:extending-contracts.adoc#using-hooks[Using Hook
                                                                   T:extending-contracts.adoc#using-hooks[Using Hook
    s].
541
                                                              541
        function _beforeTokenTransfer(address from, add
                                                                       function beforeTokenTransfer(address from, add
542
                                                              542
    ress to, uint256 amount) internal virtual { }
                                                                   ress to, uint256 amount) internal virtual { }
                                                              543 }
543 }
544
                                                              544
545 /**
                                                              545 /**
    * @title SafeERC20
                                                              546 * @title SafeERC20
^{\rm *} @dev Wrappers around ERC20 operations that throw
                                                              * @dev Wrappers around ERC20 operations that throw
    on failure (when the token
                                                                   on failure (when the token
```

function _burn(address account, uint256 amount)

497

function _burn(address account, uint256 amount)

```
alue (and instead revert or
                                                                alue (and instead revert or
    * throw on failure) are also supported, non-revert
549
                                                            * throw on failure) are also supported, non-revert
    ing calls are assumed to be
                                                                ing calls are assumed to be
550 * successful.
                                                            550 * successful.
* To use this library you can add a `using SafeERC
                                                            * To use this library you can add a `using SafeERC
    20 for IERC20; statement to your contract,
                                                                20 for IERC20; `statement to your contract,
    * which allows you to call the safe operations as
                                                            * which allows you to call the safe operations as
     `token.safeTransfer(...)`, etc.
                                                                 `token.safeTransfer(...)`, etc.
553 */
                                                            553 */
554 library SafeERC20 {
                                                            554 library SafeERC20 {
        using LowGasSafeMath for uint256;
                                                            555
                                                                  using LowGasSafeMath for uint256;
        using Address for address;
                                                                    using Address for address;
557
                                                            557
        function safeTransfer(IERC20 token, address to,
                                                                  function safeTransfer(IERC20 token, address to,
    uint256 value) internal {
                                                                uint256 value) internal {
            _callOptionalReturn(token, abi.encodeWithSe
                                                                        _callOptionalReturn(token, abi.encodeWithSe
559
    lector(token.transfer.selector, to, value));
                                                                lector(token.transfer.selector, to, value));
560
                                                            560
561
                                                            561
        function safeTransferFrom(IERC20 token, address
                                                                    function safeTransferFrom(IERC20 token, address
    from, address to, uint256 value) internal {
                                                                from, address to, uint256 value) internal {
            _callOptionalReturn(token, abi.encodeWithSe
                                                                        _callOptionalReturn(token, abi.encodeWithSe
    lector(token.transferFrom.selector, from, to, valu
                                                                lector(token.transferFrom.selector, from, to, valu
    e));
                                                                e));
                                                            564
564
        }
                                                                    }
565
                                                            565
566
                                                            566
         * @dev Deprecated. This function has issues si
                                                                     * @dev Deprecated. This function has issues si
    milar to the ones found in
                                                                milar to the ones found in
568
         * {IERC20-approve}, and its usage is discourag
                                                                     * {IERC20-approve}, and its usage is discourag
    ed.
                                                                ed.
569
                                                            569
         * Whenever possible, use {safeIncreaseAllowanc
                                                                     * Whenever possible, use {safeIncreaseAllowanc
570
                                                            570
         * {safeDecreaseAllowance} instead.
                                                                     * {safeDecreaseAllowance} instead.
                                                            571
        function safeApprove(IERC20 token, address spen
                                                                    function safeApprove(IERC20 token, address spen
    der, uint256 value) internal {
                                                                der, uint256 value) internal {
                                                            574
574
            // safeApprove should only be called when s
                                                                        // safeApprove should only be called when s
    etting an initial allowance,
                                                                etting an initial allowance,
            // or when resetting it to zero. To increas
                                                                        // or when resetting it to zero. To increas
575
    e and decrease it, use
                                                                e and decrease it, use
            // 'safeIncreaseAllowance' and 'safeDecreas
                                                                        // 'safeIncreaseAllowance' and 'safeDecreas
    eAllowance'
                                                                eAllowance'
                                                                        // solhint-disable-next-line max-line-lengt
577
            // solhint-disable-next-line max-line-lengt
                                                            577
            require((value == 0) || (token.allowance(ad
                                                                        require((value == 0) || (token.allowance(ad
578
                                                            578
    dress(this), spender) == 0),
                                                                dress(this), spender) == 0),
               "SafeERC20: approve from non-zero to no
                                                                            "SafeERC20: approve from non-zero to no
    n-zero allowance"
                                                                n-zero allowance"
580
                                                            580
            );
                                                                        );
581
            callOptionalReturn(token, abi.encodeWithSe
                                                            581
                                                                        _callOptionalReturn(token, abi.encodeWithSe
    lector(token.approve.selector, spender, value));
                                                                lector(token.approve.selector, spender, value));
582
                                                            582
                                                                  }
583
                                                            583
        function safeIncreaseAllowance(IERC20 token, ad
                                                                    function safeIncreaseAllowance(IERC20 token, ad
584
                                                            584
    dress spender, uint256 value) internal {
                                                                dress spender, uint256 value) internal {
            uint256 newAllowance = token.allowance(addr
                                                                        uint256 newAllowance = token.allowance(addr
    ess(this), spender).add(value);
                                                                ess(this), spender).add(value);
            _callOptionalReturn(token, abi.encodeWithSe
                                                                        _callOptionalReturn(token, abi.encodeWithSe
586
                                                            586
    lector(token.approve.selector, spender, newAllowanc
                                                                lector(token.approve.selector, spender, newAllowanc
    e));
                                                                e));
587
                                                            587
588
                                                            588
        function safeDecreaseAllowance(IERC20 token, ad
                                                                    function safeDecreaseAllowance(IERC20 token, ad
                                                            589
    dress spender, uint256 value) internal {
                                                                dress spender, uint256 value) internal {
```

548 * contract returns false). Tokens that return no v

548 * contract returns false). Tokens that return no v

```
590
            uint256 newAllowance = token.allowance(addr
                                                                          uint256 newAllowance = token.allowance(addr
                                                                  ess(this), spender).sub(value);
    ess(this), spender).sub(value);
            _callOptionalReturn(token, abi.encodeWithSe
                                                                          _callOptionalReturn(token, abi.encodeWithSe
591
                                                             591
                                                                  lector(token.approve.selector, spender, newAllowanc
    lector(token.approve.selector, spender, newAllowanc
592
                                                             592
                                                             593
593
594
                                                             594
595
         * @dev Imitates a Solidity high-level call (i.
                                                             595
                                                                       * @dev Imitates a Solidity high-level call (i.
    e. a regular function call to a contract), relaxing
                                                                  e. a regular function call to a contract), relaxing
    the requirement
                                                                  the requirement
596
         * on the return value: the return value is opt
                                                             596
                                                                       * on the return value: the return value is opt
    ional (but if data is returned, it must not be fals
                                                                  ional (but if data is returned, it must not be fals
597
         * @param token The token targeted by the call.
                                                                       * @param token The token targeted by the call.
598
         * @param data The call data (encoded using ab
                                                                       * @param data The call data (encoded using ab
    i.encode or one of its variants).
                                                                  i.encode or one of its variants).
                                                             599
600
        function _callOptionalReturn(IERC20 token, byte
                                                             600
                                                                      function _callOptionalReturn(IERC20 token, byte
    s memory data) private {
                                                                  s memory data) private {
            // We need to perform a low level call her
                                                                          // We need to perform a low level call her
    e, to bypass Solidity's return data size checking m
                                                                  e, to bypass Solidity's return data size checking m
    echanism, since
                                                                  echanism, since
602
            // we're implementing it ourselves. We use
                                                             602
                                                                          // we're implementing it ourselves. We use
     {Address.functionCall} to perform this call, which
                                                                   {Address.functionCall} to perform this call, which
    verifies that
                                                                  verifies that
            // the target address contains contract cod
                                                                          // the target address contains contract cod
603
    e and also asserts for success in the low-level cal
                                                                  e and also asserts for success in the low-level cal
604
                                                             604
605
            bytes memory returndata = address(token).fu
                                                             605
                                                                          bytes memory returndata = address(token).fu
    nctionCall(data, "SafeERC20: low-level call faile
                                                                  nctionCall(data, "SafeERC20: low-level call faile
                                                                  d");
606
            if (returndata.length > 0) { // Return data
                                                             606
                                                                          if (returndata.length > 0) { // Return data
    is optional
                                                                  is optional
                // solhint-disable-next-line max-line-l
                                                                              // solhint-disable-next-line max-line-l
607
                                                             607
                                                                  ength
                require(abi.decode(returndata, (bool)),
                                                                              require(abi.decode(returndata, (bool)),
608
                                                             608
     "SafeERC20: ERC20 operation did not succeed");
                                                                   "SafeERC20: ERC20 operation did not succeed");
609
            }
                                                             609
                                                                          }
610
        }
                                                             610
                                                                      }
611
                                                             611
612
                                                             612
613 interface IMEMO is IERC20 {
                                                             613 interface IMEMO is IERC20 {
        function index() external view returns ( uint
                                                                      function index() external view returns ( uint
614
                                                             614
     );
                                                                   );
615 }
                                                             615 }
616
                                                             616
                                                                  contract wsMAIA is ERC20 {
617
    contract wMEMO is ERC20 {
                                                             617
618
        using SafeERC20 for IMEMO;
                                                             618
                                                                      using SafeERC20 for IMEMO;
        using LowGasSafeMath for uint;
                                                                      using LowGasSafeMath for uint;
619
                                                             619
620
                                                             620
621
        IMEMO public immutable MEMO;
                                                             621
                                                                      IMEMO public immutable MEMO;
                                                                      event Wrap(address indexed recipient, uint256 a
        event Wrap(address indexed recipient, uint256 a
622
                                                             622
                                                                  mountMemo, uint256 amountWmemo);
    mountMemo, uint256 amountWmemo);
623
        event UnWrap(address indexed recipient, uint256
                                                             623
                                                                      event UnWrap(address indexed recipient, uint256
     amountWmemo, uint256 amountMemo);
                                                                   amountWmemo, uint256 amountMemo);
624
                                                             624
                                                                      constructor( address _MEMO ) ERC20( 'Wrapped St
625
        constructor( address _MEMO ) ERC20( 'Wrapped ME
                                                             625
                                                                   aked Maia', 'wsMAIA' ) {
     MO', 'WMEMO') {
            require( _MEMO != address(0) );
                                                                          require( _MEMO != address(0) );
626
                                                             626
            MEMO = IMEMO( MEMO);
                                                             627
                                                                          MEMO = IMEMO( MEMO);
627
628
        }
                                                             628
                                                                      }
629
                                                             629
630
                                                             630
            @notice wrap MEMO
                                                                          @notice wrap MEMO
632
            @param _amount uint
                                                             632
                                                                          @param _amount uint
                                                             633
633
            @return uint
                                                                          @return uint
634
                                                             634
```

```
635
         function wrap( uint _amount ) external returns
                                                             635
                                                                     function wrap( uint _amount ) external returns
                                                                   ( uint ) {
       ( uint ) {
                                                                        MEMO.safeTransferFrom( msg.sender, address
 636
           MEMO.safeTransferFrom( msg.sender, address
                                                             636
     (this), _amount );
                                                                  (this), _amount );
 637
                                                             637
 638
             uint value = MEMOTowMEMO( _amount );
                                                             638
                                                                          uint value = MEMOTowMEMO( _amount );
 639
             _mint( msg.sender, value );
                                                                          _mint( msg.sender, value );
 640
             emit Wrap(msg.sender, _amount, value);
                                                             640
                                                                          emit Wrap(msg.sender, _amount, value);
 641
             return value;
                                                             641
                                                                          return value;
                                                             642
 642
         }
                                                                      }
 643
                                                             643
 644
                                                             644
 645
             @notice unwrap MEMO
                                                             645
                                                                          @notice unwrap MEMO
             @param _amount uint
                                                                          @param _amount uint
             @return uint
                                                                          @return uint
 648
                                                             648
         function unwrap( uint _amount ) external return
                                                                      function unwrap( uint _amount ) external return
 649
     s ( uint ) {
                                                                  s ( uint ) {
 650
             _burn( msg.sender, _amount );
                                                             650
                                                                          _burn( msg.sender, _amount );
 651
                                                             651
             uint value = wMEMOToMEMO( _amount );
                                                             652
                                                                          uint value = wMEMOToMEMO( _amount );
 652
             MEMO.safeTransfer( msg.sender, value );
                                                                          MEMO.safeTransfer( msg.sender, value );
             emit UnWrap(msg.sender, _amount, value);
                                                             654
                                                                          emit UnWrap(msg.sender, _amount, value);
 655
             return value:
                                                             655
                                                                          return value;
                                                             656
 656
         }
                                                                      }
 658
                                                             658
 659
             @notice converts wMEMO amount to MEMO
                                                             659
                                                                          @notice converts wMEMO amount to MEMO
                                                             660
                                                                          @param _amount uint
             @param _amount uint
 661
             @return uint
                                                             661
                                                                          @return uint
 662
                                                             662
         function wMEMOToMEMO( uint \_amount ) public vie
                                                                      function wMEMOToMEMO( uint _amount ) public vie
     w returns ( uint ) {
                                                                  w returns ( uint ) {
664
         return _amount.mul( MEMO.index() ).div( 10
                                                                      return _amount.mul( MEMO.index() ).div( 10
                                                             664
       ** decimals );
                                                                   ** decimals );
 665
         }
                                                             665
                                                                      }
 666
                                                             666
 667
                                                             667
             @notice converts MEMO amount to wMEMO
                                                                          @notice converts MEMO amount to wMEMO
 668
                                                             668
 669
             @param _amount uint
                                                             669
                                                                          @param _amount uint
 670
             @return uint
                                                             670
                                                                          @return uint
 671
                                                             671
                                                                   function MEMOTowMEMO( uint _amount ) public vie
       function MEMOTowMEMO( uint _amount ) public vie
     w returns ( uint ) {
                                                                  w returns ( uint ) {
            return _amount.mul( 10 ** decimals ).div( M
                                                                          return _amount.mul( 10 ** decimals ).div( M
                                                                  EMO.index() );
     EMO.index());
                                                             674
 674
                                                             675
 675
 676 }
                                                             676 }
```